NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

ANALYSIS OF A PROPOSED THIRD GENERATION (3G) MOBILE COMMUNICATION STANDARD, TIME DIVISION – SYNCHRONOUS CODE DIVISION MULTIPLE ACCESS (TD-SCDMA)

by

Donald H. Paulson, Jr.

June 2002

Thesis Advisor:	R. Clark Robertson
Co-Advisor:	Tri T. Ha

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE			*Form Approved OMB No. 0704-0188*

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	June 2001	Master's Thesis

4. TITLE AND SUBTITLE: Title (Mix case letters)
Analysis of a proposed Third Generation (3G) Mobile Communication Standard, Time Division – Synchronous Code Division Multiple Access (TD-SCDMA)

5. FUNDING NUMBERS

6. AUTHOR(S) Donald H Paulson, Jr.

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Naval Postgraduate School
Monterey, CA 93943-5000

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING /MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Nation Security Agency, Applied Technologies Division
9800 Savage Rd. Fort George G. Meade, MD 20755

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

12a. DISTRIBUTION / AVAILABILITY STATEMENT
Approved for public release; distribution is unlimited

12b. DISTRIBUTION CODE

13. ABSTRACT *(maximum 200 words)*

With a growing number of consumers utilizing the Internet, companies have foreseen a consumer demand for high-speed wireless access. Since current mobile cellular systems can transfer at most 115.2 kbps per user, a third generation of mobile cellular service has been under development by various organizations since 1997. This new generation of technology will support data rates up to 2 Mbps for stationary mobiles and up to 144 kbps for vehicular traffic.
This thesis focuses mainly on TD-SCDMA, one of many candidates submitted to the International Telecommunications Union for third generation review. The standard, developed in China by the Chinese Academy of Telecommunications Technology, employs both code-division multiple access and time-division duplexing to support both forward and reverse transmissions on one physical layer. This aspect, along with other common features of TD-SCDMA, will be studied and evaluated to determine if this new technology is a viable option for future commercial or military deployment.

14. SUBJECT TERMS: Time Division Synchronous Code Division Multiple Access (TD-SCDMA), 3G Cellular Communications, Raised Root Cosine Filtering, CDMA, Orthogonal Variable Spreading Factor (OVSF), Time Division Duplexing (TDD)

15. NUMBER OF PAGES
96

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	UL

NSN 7540-01-280-5500 Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. 239-18

THIS PAGE INTENTIONALLY LEFT BLANK

ANALYSIS OF A PROPOSED THIRD GENERATION (3G) MOBILE COMMUNICATION STANDARD, TIME DIVISION – SYNCHRONOUS CODE DIVISION MULTIPLE ACCESS (TD-SCDMA)

Donald H. Paulson, Jr.
Ensign, United States Navy
B.S., University of Nebraska - Lincoln, 2001

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN ELECTRICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL
June 2002

Author: Donald H. Paulson, Jr.

Approved by: R. Clark Robertson
Thesis Advisor

Tri T. Ha
Co-Advisor

Jeffrey B. Knorr
Chairman, Department of Electrical and Computer Engineering

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

With a growing number of consumers utilizing the Internet, companies have foreseen a consumer demand for high-speed wireless access. Since current mobile cellular systems can transfer at most 115.2 kbps per user, a third generation of mobile cellular service has been under development by various organizations since 1997. This new generation of technology will support data rates up to 2 Mbps for stationary mobiles and up to 144 kbps for vehicular traffic.

This thesis focuses mainly on TD-SCDMA, one of many candidates submitted to the International Telecommunications Union for third generation review. The standard, developed in China by the Chinese Academy of Telecommunications Technology, employs both code-division multiple access (CDMA) and time-division duplexing (TDD) to support both forward and reverse transmissions on one physical layer. This aspect, along with other common features of TD-SCDMA, will be studied and evaluated to determine if this new technology is a viable option for future commercial or military deployment.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

THIS PAGE INTENTIONALLY LEFT BLANK

EXECUTIVE SUMMARY

The purpose of this thesis is to provide an independent evaluation of a proposed third generation (3G) standard for cellular communications, Time Division Synchronous Code Division Multiple Access (TD-SCDMA). With a growing number of consumers utilizing the Internet, companies have foreseen a consumer demand for high-speed wireless access. Since current mobile cellular systems can transfer at most 115.2 kbps per user and are costly to the service provider, a third generation of mobile cellular service has been under development since 1997. This new generation of technology will support data rates up to 2 Mbps for stationary mobiles and up to 144 kbps for vehicular traffic.

TD-SCDMA is a joint venture between the Chinese Academy of Telecommunications Technology (CATT) and Siemens Information and Communication Mobile Group (Siemens IC Mobile). The concept for TD-SCDMA was originally submitted to the International Telecommunications Union (ITU) as a separate candidate submission for IMT-2000, but since then has also been incorporated into the Universal Terrestrial Radio Access - Time Division Duplex (UTRA TDD) proposal. The Chinese Ministry of Information Industry and Chinese Wireless Telecommunication Standard group (CWTS) were also instrumental in submitting this new technology for international review, but most of the technical information was originated by the previous two sources.

TD-SCDMA is designed to utilize the Global System for Mobile (GSM) core network architecture, which is a second generation (2G) technology using time-division multiple access (TDMA) and Gaussian minimum-shift keying (GMSK). GSM is currently the world's largest service provider for cellular communications, dominating 62% of the market with currently over 600 million subscribers. By being backward compatible with this network, TD-SCDMA will have a great marketing advantage over non-GSM based technology. Currently in the U.S., the cellular service providers AT&T and Cingular use GSM networks and have announced their support for Universal Mobil Telecommunications System (UMTS), which includes WCDMA (Wideband CDMA) and

TD-SCDMA. In other countries, Europe uses the GSM network almost exclusively and Asia provides another large market for 3G technologies.

The largest marketing opportunity for TD-SCDMA is in China where the new technology was first developed. By December 2001, China had become the world's largest mobile telephone market with over 140 million subscribers. This number is staggering considering there was only a 7% market penetration, and China has the potential to grow to an estimated 400 million users by 2007. CATT and Siemens IC Mobile are currently expected to deploy TD-SCDMA there as early as 2003.

As the name implies, TD-SCDMA utilizes time-division duplexing (TDD) along with synchronous CDMA to multiplex and spread a baseband signal. The TDD aspect allows one user several time slots for either uplink or downlink transmission, and the CDMA aspect allows multiple users to share the same physical channel (1.6 MHz bandwidth) and hence, time slot. As a comparison with non-TDD systems, IS-95 users need access to two physical channels (two different frequency bands) to obtain both uplink and downlink transmission, occupying a total bandwidth of 2.5 MHz.

When analyzed, the performance of TD-SCDMA under adverse conditions is very similar to other CDMA systems. A slight difference lies in the fact that TD-SCDMA employs matching root-raised cosine filters at both the transmitter and receiver to reduce inter-symbol interference. Interestingly, the author found that having this type of filter at the receiver has minimal effect on additive white Gaussian noise (AWGN), even though the filter is low-pass in nature. This phenomenon was computed analytically and verified via simulation. Implementation of this property made further analysis much easier and allowed the author to use existing analytical equations to verify the simulations.

Because AWGN is not an interesting or challenging adverse condition, Rayleigh fading was also considered and additional simulation results were produced. Signal fading is defined as amplitude variations in a received signal due to a time-varying multipath channel. As we saw with AWGN, the simulation results were similar to theoretical results since TD-SCDMA is akin to most direct sequence spread spectrum systems.

The next step was an attempt to disrupt the simulated TD-SCDMA transmission. Two types of interfering methods were employed, tone jamming and barrage-jamming. Because the performance of TD-SCDMA is similar to other CDMA systems, the results of jamming the transmission were very close. Only the tone-jamming scenario produced significantly different results, but this was due to the fact that TD-SCDMA employs very small spreading factors and the only theoretical equations that exist are for large spreading factors. We found that for small spreading factors the tone-jamming signal has an effect similar to a standard non-CDMA quadrature phase shift-keying (QPSK) signal with tone jamming. By adding a constant related to the spreading factor, the author was able to match an analytical equation to the numerical simulations. This could not be verified for all spreading factors, but enough compared relatively well that these theoretical equations have some merit.

The last stage in analysis was to theorize how an individual or organization could intercept and exploit a TD-SCDMA transmission. Because all 3G standards are published and well documented, the radio protocol processes can be adapted to allow for covert interception of TD-SCDMA transmissions. The key to the process is synchronization, and without this fundamental aspect no detection can take place. Once this synchronization is achieved, with both the base station and the intended target, then more detailed procedures can theoretically be developed to intercept and interpret the signals.

All things considered, despite TD-SCDMA's advantage by using TDD and having a smaller bandwidth, the author does not foresee this technology gaining a large share of the 3G market unless China goes with their homegrown system. WCDMA and CDMA2000, the two main competitors, already have a solid core network and marketing base in various parts of the world. This means that TD-SCDMA, which uses principles employed by both WCDMA and CDMA2000, will have a hard time attracting customers. This can be seen in the fact that TD-SCDMA was incorporated in UTRA as a low chiprate *option* whereas WCDMA is more prominent in that standard. Without a key feature that will improve the performance of a system above and beyond WCDMA or CDMA2000, the author doubts any other system will gain much popularity.

THIS PAGE INTENTIONALLY LEFT BLANK

I. INTRODUCTION

A. PURPOSE

The purpose of this thesis is to provide an independent evaluation of a proposed Third Generation (3G) standard for cellular communications. The main focus will be on a concept developed in China entitled Time Division – Synchronous Code Division Multiple Access (TD-SCDMA). From an examination of this standard, ways to exploit, intercept, and block future transmissions can be researched and developed.

B. BACKGROUND OF CELLULAR STANDARDS

Cellular communications, since its commercial introduction in the United States in 1983, has undergone many changes to keep pace with advancing technology. Initially, AT&T developed the U.S. Advanced Mobile Phone System (AMPS) utilizing frequency modulation (FM) and frequency-division multiple access (FDMA) for multiple user access. Much thought and field testing had been put into releasing this standard since AT&T and Bell Laboratories had been researching and developing cellular technology as far back at 1958 [1]. Essentially, all future developments have relied heavily upon these results. With the rate of changing technology today, few companies can afford the amount of field-testing and research that was conducted by this company.

A second generation of cellular communications, introduced in 1991, is the U.S. Digital Cellular (USDC), commonly called IS-54 (Interim Standard - 54). Instead of using FM and FDMA, IS-54 utilizes $\pi/4$-DQPSK digital modulation and time-division multiple access (TDMA) for multiple user access. This was quite a departure from the AMPS standard, which uses analog signaling. For the same frequency spectrum and channel bandwidth, IS-54 has three times the user capacity of AMPS [2].

Soon after the inception of IS-54, a new standard was developed using similar digital technology. IS-95, commonly called CDMA (code-division multiple access), was introduced in 1993 and heralded in a new age for cellular communications. Whereas previous systems required cellular cluster planning and channel reuse schemes, CDMA required very little of this. CDMA uses Walsh functions, which are orthogonal to each other, and pseudorandom sequences to spread the spectrum of the transmitted signal.

Because these sequences are orthogonal to each other, multiple users can use the same frequency band. A receiver can extract the desired signal if it has the proper code, and the orthogonality of the other sequences cause the interference to be almost zero.

With a growing number of consumers utilizing the Internet, companies have foreseen a consumer demand for high-speed wireless access. Since current mobile cellular systems can transfer at most 115.2 kbps per user (IS-95B) [2] and is costly to the service provider, a third generation of mobile cellular service has been under development since 1997. The new generation of technology will support data rates up to 2 Mbps for stationary mobiles and up to 144 kbps for vehicular traffic [3]. Of the many proposed standards, this thesis will mainly cover TD-SCDMA.

C. PROPONENTS OF TD-SCDMA

TD-SCDMA is a joint venture between the Chinese Academy of Telecommunications Technology (CATT) and Siemens Information and Communication Mobile Group (Siemens IC Mobile). The concept for TD-SCDMA was originally submitted to the International Telecommunications Union (ITU) as a separate candidate submission for IMT-2000, but since then has also been incorporated into the Universal Terrestrial Radio Access - Time Division Duplex (UTRA TDD) proposal. The Chinese Ministry of Information Industry and Chinese Wireless Telecommunication Standard group (CWTS) were also instrumental in submitting this new technology for international review, but most of the technical information was originated by the previous two sources.

As stated in their objectives, the IMT-2000 project was instituted to promote support for harmonizing international frequency spectrums and developing compatible mobile telecommunications systems. This goal has not yet been fully realized, but the international community has narrowed development of 3G technologies into five distinct groups. Figure 1.1 illustrates the five main proposals and how they are distinct from one another.

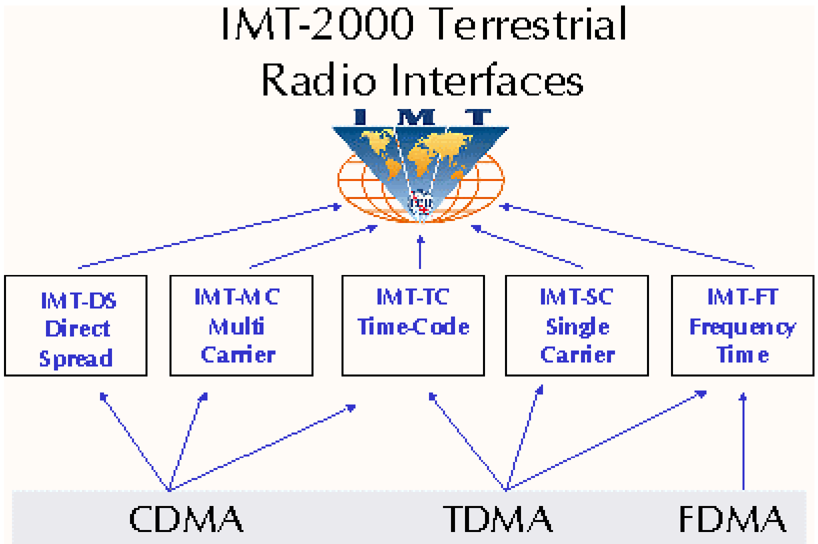

Figure 1.1. IMT-2000 Terrestrial Radio Interfaces. The radio interfaces shown are commonly known by the following names: WCDMA for IMT-DS; CDMA2000 for IMT-MC; UTRA TDD, and TD-SCDMA for IMT-TC; UWC-136 for IMT-SC; and DECT for IMT-FT. From Ref. [4].

The 3rd Generation Partnership Project (3GPP) currently holds the most recent specifications for 3G standards based on the GSM core network. All technical information contained within this thesis was obtained from this source. 3GPP was formed in 1998 and maintains all technical specifications for UTRA FDD, UTRA TDD (including TD-SCDMA), WCDMA, GPRS, EDGE, and GSM. A second project, 3GPP2, was instituted at the same time and deals exclusively with CDMA2000 and ANSI/TIA/EIA-41.

D. WHO WILL USE TD-SCDMA?

TD-SCDMA is designed to utilize the Global System for Mobile (GSM) core network architecture, which is a 2G technology using TDMA and Gaussian minimum-shift keying (GMSK). GSM is currently the world's largest service provider for cellular communications, dominating 62% of the market with currently over 600 million subscribers [5]. By being backward compatible with this network, TD-SCDMA will have a great marketing advantage over non-GSM based technology (Figure 1.2 illustrates the evolution from current 2G systems to 3G). Currently in the U.S., the cellular service providers AT&T and Cingular use GSM networks and have announced their support for Universal Mobil Telecommunications System (UMTS), which includes WCDMA

(Wideband CDMA) and TD-SCDMA. In other countries, Europe uses the GSM network almost exclusively and Asia provides another large market for 3G technologies.

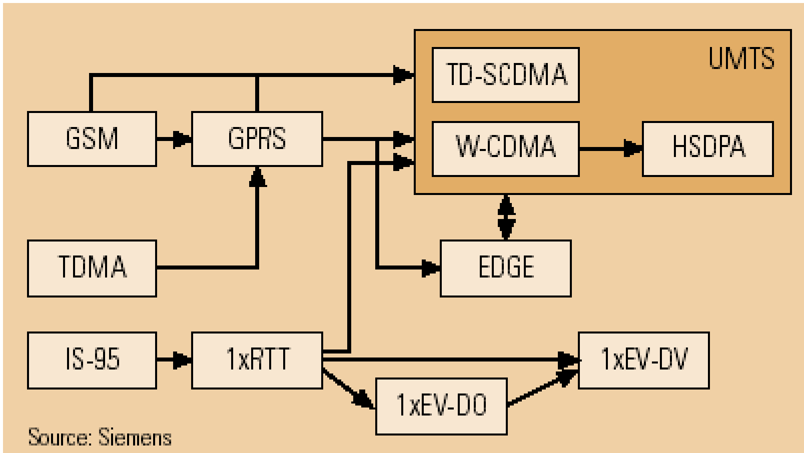

Figure 1.2. The road from 2G to 3G. From Ref. [5].

The largest marketing opportunity for TD-SCDMA is in China where the new technology was first developed. By December 2001, China had become the world's largest mobile telephone market with over 140 million subscribers [6]. This number is staggering considering there was only a 7% marketing penetration, and China has the potential to grow to an estimated 400 million users by 2007 [6]. CATT and Siemens IC Mobile are currently expected to deploy TD-SCDMA there as early as 2003.

E. ORGANIZATION OF STUDY

The remainder of this thesis will discuss, in general, an overview and evaluation of TD-SCDMA. All technical information referenced in this document was obtained using the 3GPP standards dealing with UTRA-TDD (low chip rate option), which is based solely on TD-SCDMA technology. Chapters II introduces the fundamentals of TD-SCDMA that make it both similar and different from current second generation and other third generation standards. Topics to be covered are the physical layer, transmission and reception, and a comparison of this standard with two other 3G proposals. Chapter III presents an analysis of TD-SCDMA. This section evaluates the performance of the system under the unfavorable conditions of ambient noise and jamming. From there, Chapter IV will explore two methods of signal interception and exploitation, while Chapter V presents the writer's conclusions and recommendations.

II. FUNDAMENTALS OF TD-SCDMA

A. PHYSICAL LAYER

1. General Description

As the name implies, TD-SCDMA utilizes time-division duplexing (TDD) along with synchronous CDMA to multiplex and spread a baseband signal. The TDD aspect allows one user several time slots for either uplink or downlink transmission, and the CDMA aspect allows multiple users to share the same physical channel and, hence, time slot. As a comparison with non-TDD systems, IS-95 users need access to two physical channels (two different frequency bands) to obtain both uplink and downlink transmission, occupying a total bandwidth of 2.5 MHz. For TD-SCDMA, each physical channel can provide both uplink and downlink capabilities, occupying only 1.6 MHz/carrier. (Note: for the remainder of this thesis all references to the physical channel imply the actual 1.6MHz frequency spectrum bandwidth occupied by the transmitted information) With the auction of radio frequency spectrums generating bids in the millions of dollars, a 43% saving in user bandwidth is significant.

Another key feature of TD-SCDMA is the ability to support information data rates of 12.2, 64, 144, 384, and 2048 kbps. Except in the case of 2048 kbps, individual users can achieve higher data rates by being assigned multiple CDMA codes. Alternatively, in the case of 2048 kbps no CDMA spreading is used and *this is only a downlink capability and cannot be used for uplink*. As previously mentioned, the current 2.5G cellular communications technology only supports data rates up to 115.2 kbps by using the same technique, but because IS-95B requires two physical channels this costs almost twice the bandwidth of one TD-SCDMA two-way channel. This also significantly reduces the number of users/cell available in IS-95B.

2. Spreading and Modulation

Spreading of TD-SCDMA is similar to other CDMA systems in that TD-SCDMA utilizes orthogonal codes to allow multiple users on the same physical channel. For this standard, a variable sequence of up to sixteen orthogonal Walsh codes and a set of cell-specific scrambling codes is applied to a data sequence to spread the information data's

spectrum. Because the orthogonality of Walsh codes is destroyed in a multipath environment [7], this requires TD-SCDMA to maintain both uplink and downlink phase and timing synchronization. Being time aligned is very important for CDMA, and without synchronization TD-SCDMA will not work.

TD-SCDMA also employs forward error correction (FEC) coding and the modulation techniques of QPSK and 8PSK to support data rates up to 2048 kbps. The most recent standard publication mentions an additional modulation technique of 16QAM, but this aspect is not fully discussed in the documentation. Table 2.1 illustrates the MPSK complex symbol representations currently used for modulation.

Table 2.1. Complex symbol representation for QPSK and 8PSK modulation.

QPSK		**8PSK**	
Consecutive binary bit pattern	Complex symbol $\underline{d}_n^{(k,i)}$	Consecutive binary bit pattern	Complex symbol $\underline{d}_n^{(k,i)}$
00	+j	000	cos(11π/8)+ jsin(11 π /8)
01	+1	001	cos(9 π /8)+ jsin(9 π /8)
10	-1	010	cos(5 π /8)+ jsin(5 π /8)
11	-j	011	cos(7 π /8)+ jsin(7 π /8)
		100	cos(13 π /8)+ jsin(13 π /8)
		101	cos(15 π /8)+ jsin(15 π /8)
		110	cos(3 π /8)+ jsin(3 π /8)
		111	cos(π /8)+ jsin(π /8)

a. Orthogonal Variable Spreading Factors

To allow multiple users on the same physical channel without causing multi-user interference, each data waveform is spread by an orthogonal channelization code. This channelization code is generated from a set of Orthogonal Variable Spreading Factor (OVSF) codes and keeps the correlation of multiple signals on the same physical channel low. Without orthogonal coding, multiple signals on the same physical channel would interfere with each other and significantly increase the probability of bit error. By employing an orthogonal coding scheme and maintaining the same transmitted power for all users, multiple signals can be on the same physical channel and not interfere with each other. Figure 2.1 illustrates the code-tree for OVSF.

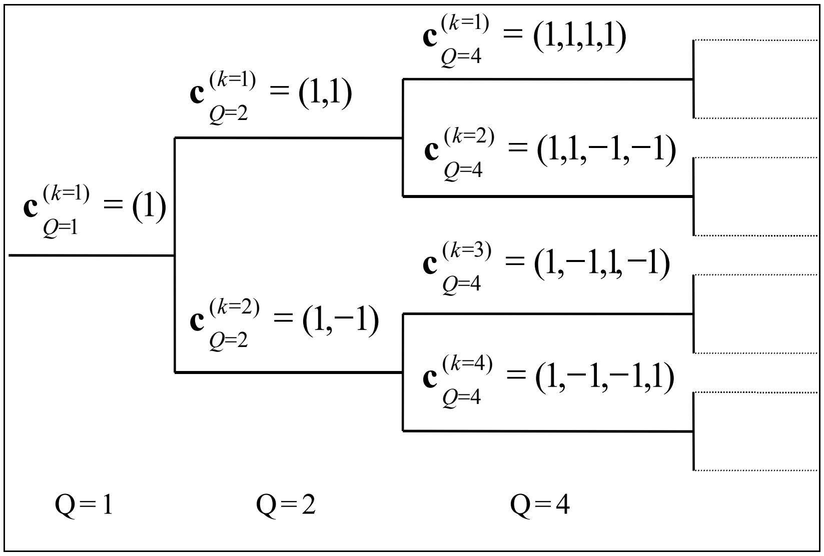

Figure 2.1. Channelization code tree for Orthogonal Variable Spreading Factor (OVSF) generation. After Ref. [8].

In Figure 2.1, the vector c is the specific channel code, k is the Walsh code number, and Q is the spreading factor (SF) where $Q \in \{1, 2, 4, 8, 16\}$. As the orthogonal tree branches from left to right the spreading factor increases, the supported information data rates decrease, and the number users able to access the same physical channel also decrease. Table 2.2 defines the number of information data symbols that can be transmitted with a specific spreading factor in one burst transmission. Notice that the smaller the spreading factor the more symbols can be transmitted, but keep in mind that the number of users per physical channel also decreases.

Table 2.2. Number of symbols per burst transmission.

Spreading Factor (Q)	**Number of symbols per burst transmission**
1	352
2	176
4	88
8	44
16	22

For uplink, OVSF works by assigning either single or multiple codes at various spreading factors to each user based on the number of users on a particular physical channel and the data rate that is requested by each user. In contrast, for the

downlink TD-SCDMA only allows spreading factors of Q=1 and Q=16, but can still assign single or multiple codes to each user. As an example, for voice communications a data rate of only 12.2 kbps is required and up to sixteen users can be supported on one physical channel by using a spreading factor of Q=16. For high speed internet access on the downlink or down-streaming video at 2048 kbps, TD-SCDMA switches to 8PSK and a spreading factor of Q=1, which implies no spreading at all. Section B of this chapter will discuss the time frame allocation and information data rates in more detail.

Since TD-SCDMA can support data rates *up to* 2048 kbps, the code tree represents a dynamic system that changes the spreading factor as the number of users/channel and requested data rates vary. For example, if two users were sharing the same channel and each requested the fastest possible data rate, the system could choose a spreading factor of two and the appropriate branches (see Figure 2.2.).

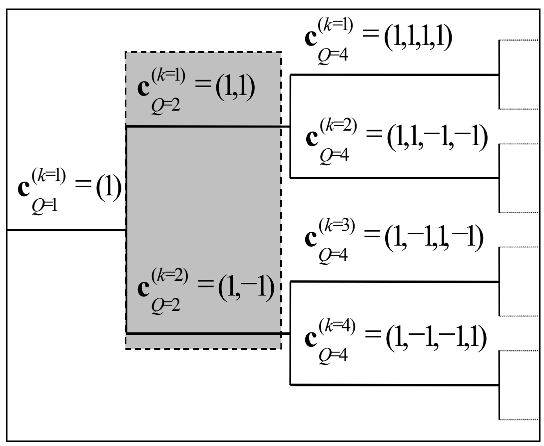

Figure 2.2. Example of two users on the same channel utilizing the fastest possible data rate and different OVSF codes. After Ref. [8].

As a general rule, to maintain the orthogonality of the user codes, no assigned code can trace its way to the root of the tree through another code that is already in use. For example, if an additional user were assigned this same physical channel the system would have to reconfigure. One way to accomplish this would be to keep one user at the faster data rate and support the other two users on a lower data rate as shown in Figure 2.3(a). Another suitable option would be to move all three users to a lower data rate as shown in Figure 2.3(b). Both of these techniques are perfectly acceptable. In contrast, an incorrect choice would be to choose a configuration as shown in Figure 2.3(c). This last figure shows that one of the channelization codes chosen is the

derivative of another. In other words, one code can trace a path to the root of the tree through another code already in use. This means that the orthogonality of the two signals is lost and multi-user interference will occur between the two users sharing the same code tree branch. In practice, TD-SCDMA prefers to use a spreading factor of Q=16 on the downlink and assign multiple codes to the same user to achieve faster data rates. An exemption to this preference is in the case of Q=1 and 8PSK, which only allows one user on a physical channel.

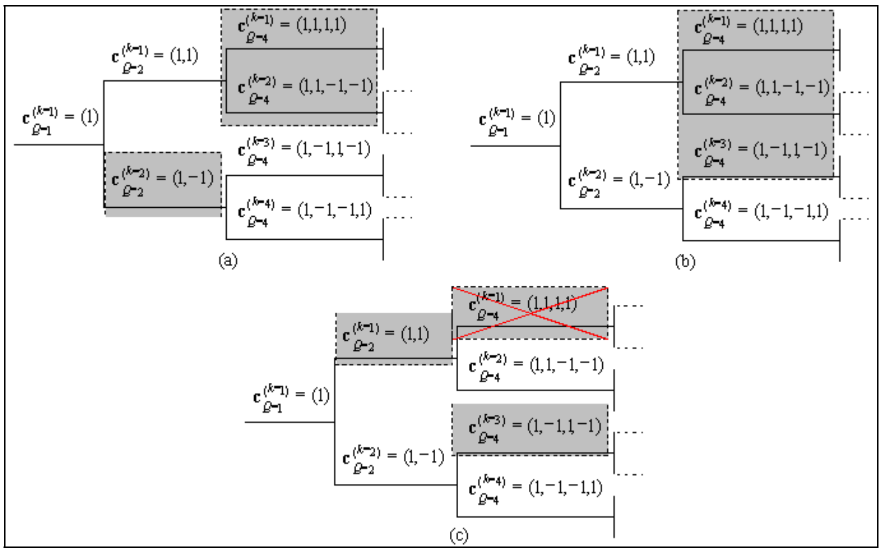

Figure 2.3. Different channelization coding to allow three users access on the same channel. Case (a) and (b) are correct examples while case (c) would cause multi-user interference. After Ref. [8].

b. Cell-Specific Scrambling Codes

After channelization coding, each complex code is multiplied by a code-specific multiplier $w_{Q_k}^{(k)}$, where

$$w_{Q_k}^{(k)} = e^{\frac{j\pi}{2} p_k}, p_k = \{0, \ldots, Q_{k-1}\}, \quad (2.1)$$

and a complex scrambling code $\underline{v}$. The standard is not clear as to the purpose of the code-specific multiplier $w_{Q_k}^{(k)}$, but the scrambling code $\underline{v}$ is cell-specific and each user within a given cell shares the same scrambling code. Because orthogonality is obtained from the OVSF codes, the scrambling code allows a mobile to distinguish the desired

base station signals from adjacent base station transmissions. By definition, the scrambling codes are always of length sixteen and are taken from the complex set:

$$\underline{v} = \{\underline{v}_1, \underline{v}_2, ..., \underline{v}_{16}\} \qquad \underline{v}_i = j^i v_i \qquad v_i \in \{1,-1\}, i = 1,...,16 \,. \tag{2.2}$$

Combining the user specific channelization code and cell specific scrambling code, we get the following equation:

$$s_p^{(k)} = c^{(k)}_{1+[(p-1)\bmod Q_k]} \bullet \underline{v}_{1+[(p-1)\bmod Q_{MAX}]}, k = 1,...K_{Code}, p = 1,..., N_k Q_k \,, \tag{2.3}$$

where N_k is the number of encoded data bits per time slot and K_{Code} is the total number of users on the channel. To allow for variable length channelization codes, the equation uses the *modulo* operator so the OVSF code and cell-specific scrambling code can overlap and repeat themselves. This ensures that a mobile is able to constantly identify the base station even if the variable spreading factor changes.

c. Baseband Spread Signal

Applying all the individual components from the previous sections, we find that the encoded data is spread according to the following formula:

$$d^{(k,i)} = \sum_{n=1}^{N_k} d_n^{(k,i)} w_{Q_k}^{(k)} \sum_{q=1}^{Q_k} s_{(n-1)Q_k+q}^{(k)} \bullet$$

$$Cr_0(t-(q-1)T_c-(n-1)Q_kT_c-(i-1)(N_kQ_kT_c+L_mT_c)), i = 1,2 \,, \tag{2.4}$$

where $d^{(k,i)}$ is the transmitted complex-valued chip sequence, $d_n^{(k,i)}$ is the encoded user information data using FEC coding, and Cr_0 is the impulse response for a root-raised cosine (RRC) filter. The index *i* is used to signify that the data sequence in one timeslot is divided into sections, and the reason for this will be explained in Section B. The purpose of the root-raised cosine filter will be discussed in more detail in the next section.

3. Passband Modulation

To transmit the baseband chip sequence described in the previous section, TD-SCDMA uses an IQ modulator as shown in Figure 2.4. This modulator splits the complex chip sequence into its real and imaginary parts and pulse shapes the complex data impulses using identical root-raised cosine filters.

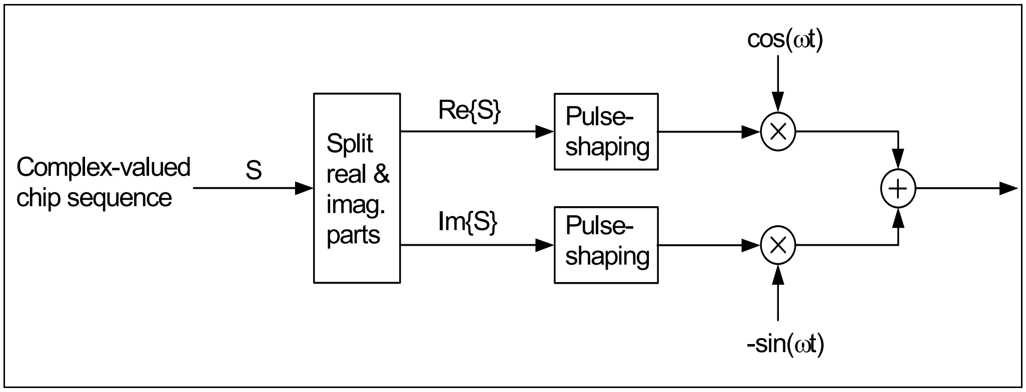

Figure 2.4. Modulation of baseband complex-valued chip sequence using raised root cosine filters for pulse shaping and an IQ modulator for heterodyning. From Ref. [8].

The root-raised cosine filters are implemented to reduce inter-symbol interference (ISI) in the channel by following Nyquist's pulse-shaping criterion [9]. Since TD-SCDMA is restricted in bandwidth to 1.6 MHz, any signal energy that spills over into adjacent frequency bands will cause interference. The principle behind raised cosine filtering is that the frequency response of the filter is essentially flat over the desired frequency band, has a sharp transition at the cutoff frequency, and is essentially zero in the stopband. Figure 2.5 illustrates the frequency response of a raised cosine filter, and the transfer function is given by

$$H_{RC}(f)=\begin{cases} 1 & 0\le |f| \le \dfrac{(1-\alpha)}{2T_c} \\ \dfrac{1}{2}\left[1+\cos\left[\dfrac{\pi(|f|\bullet 2T_s-1+\alpha)}{2\alpha}\right]\right] & \dfrac{(1-\alpha)}{2T_c}\le |f| \le \dfrac{(1+\alpha)}{2T_c} \\ 0 & |f|\le \dfrac{(1+\alpha)}{2T_c} \end{cases}. \tag{2.5}$$

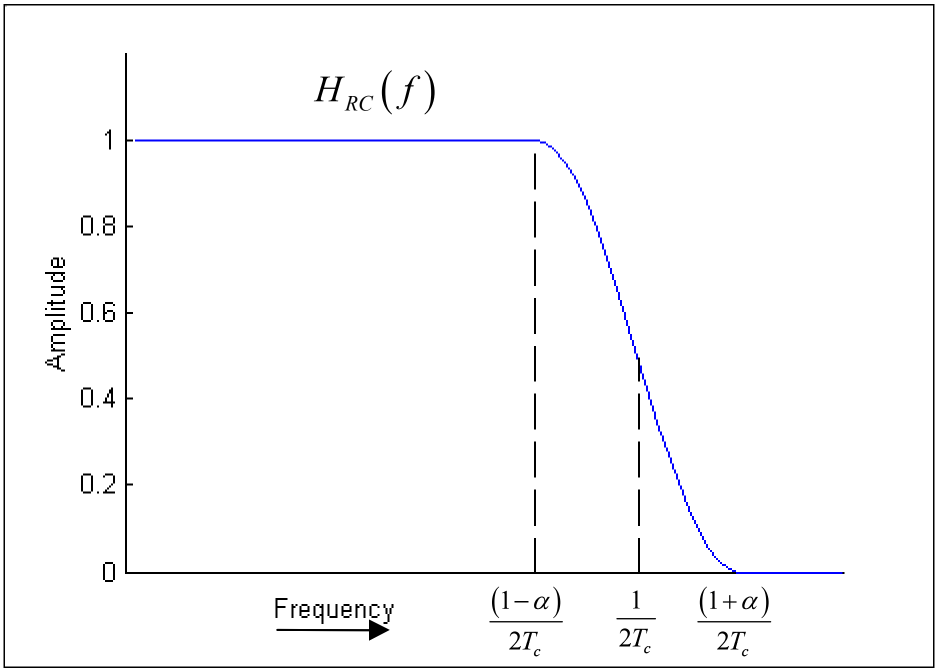

Figure 2.5. Frequency response of raised cosine filter. The value of T_c is the chip time and α is the roll-off factor.

By choosing an appropriate roll-off factor α, we can limit the amount of spillover. Since TD-SCDMA uses a chiprate of 1.28 Mcps, by applying a rolloff factor of α=0.22 we can limit the baseband spectrum to ±0.7808 Mcps and a total passband bandwidth of 1.5616 Mcps. This is the reason why TD-SCDMA matches the 1.28 Mcps chiprate with a bandwidth of 1.6 MHz.

By placing matched RRC filters at the receiver and transmitter, we effectively create a raised cosine (RC) filter at the receiver. The only drawback of implementing raised cosine filters is that since the frequency response of the filter is almost a rectangular pulse for small α, the time response is similar to a sinc function $\left(\frac{\sin(x)}{x}\right)$. This is taken from the fact that the inverse Fourier transform of a unit-step function in the frequency domain is a sinc function in the time domain. The problem lies in the fact that a sinc function is not physically realizable since the waveform is a non-casual function (the response at any point in time is dependent on both past and future inputs) and exists for all time (-∞, ∞). The standard procedure is to terminate the impulse response three time units before and after t=0 and delay the output three time units to make the function

causal. Figure 2.6 illustrates the impulse response for the filter described above (before delaying the signal three time units), and the impulse response is given by

$$Cr_0 = \frac{\sin\left(\pi \frac{t}{T_c}(1-\alpha)\right) + 4\alpha \frac{t}{T_c} \cos\left(\pi \frac{t}{T_c}(1+\alpha)\right)}{\pi \frac{t}{T_c}\left(1-\left(4\alpha \frac{t}{T_c}\right)^2\right)}, \quad \alpha = 0.22 . \tag{2.6}$$

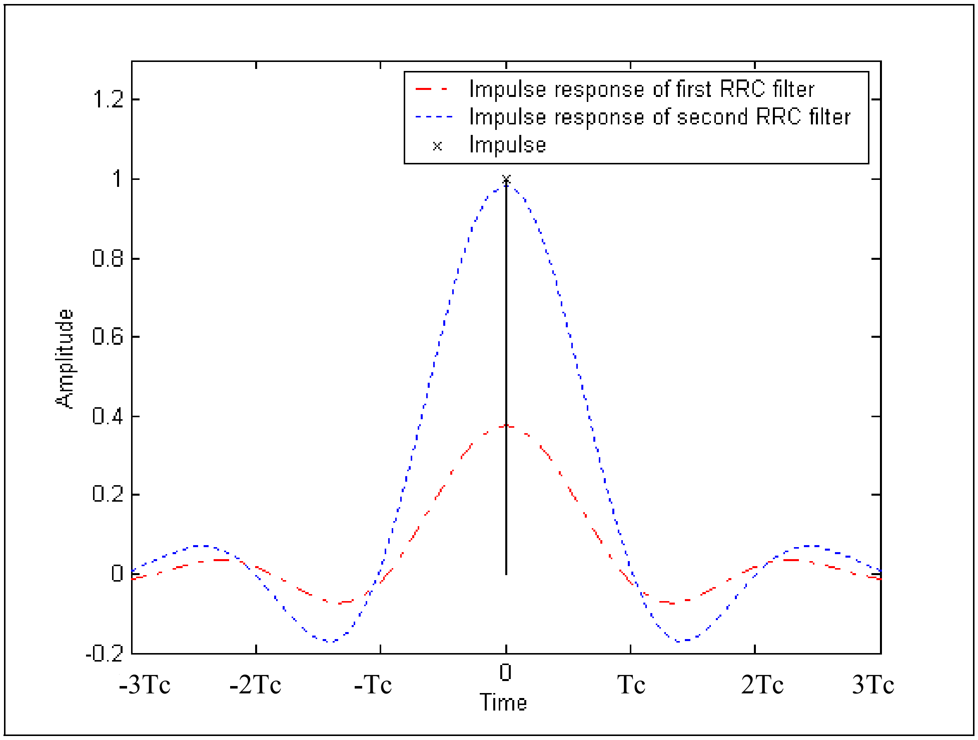

Figure 2.6. The impulse of response of a root-raised cosine filter (dot-dashed line) after being hit by an impulse (solid line). We can achieve the impulse response of a raised cosine filter (dotted line) if we pass the first waveform through a second RRC filter. In this case α=0.22.

By inspecting Figure 2.6, we see that passing the first RRC waveform through a matched RRC filter produces the output of a single RC filter. This is instrumental in reproducing the original input data stream at the receiver. The key to RC filtering is that the nulls of the waveform occur every T_c seconds. If two impulses occurring T_c seconds apart were passed through a raised cosine filter, the result would be as shown in Figure 2.7. The actual output of the RC filter is taken by summing the individual impulse responses from the two inputs. Since the nulls of the two separate waveforms occur every T_c seconds, this means that the two signals will have no interference at the chip

time T_c. By sampling the waveform every T_c seconds, the original impulse train can be reconstructed. This is how the raised cosine filter prevents ISI.

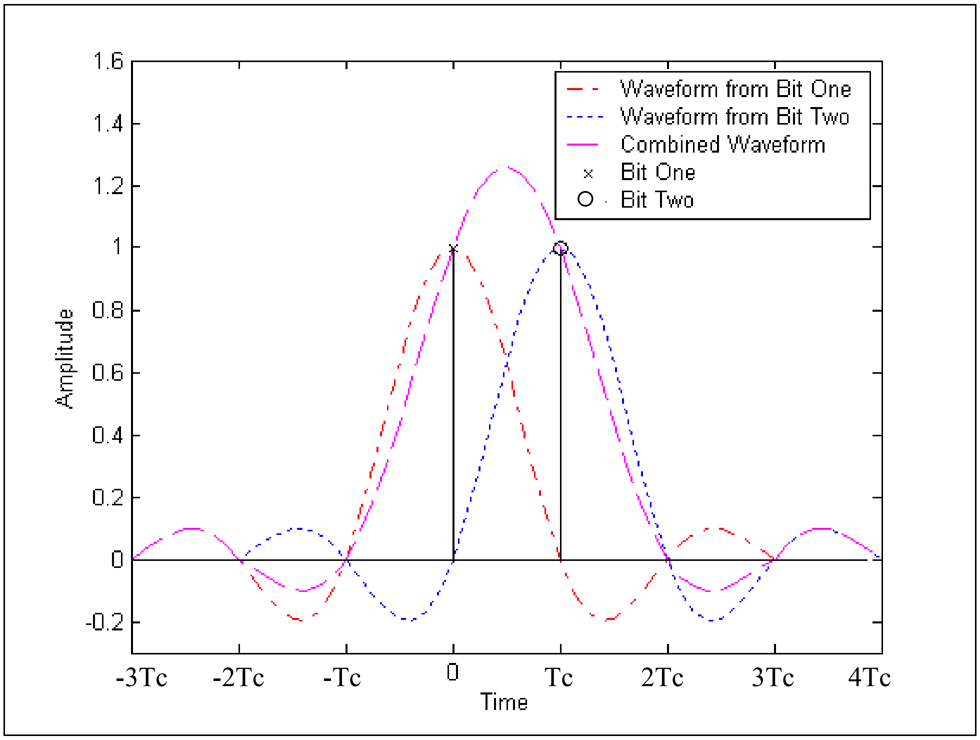

Figure 2.7. The output of a raised cosine filter after two incoming data bits. The impulse response after the first bit (dot-dashed line) and the second bit (dotted line) are combined to create the actual output waveform (dashed line).

B. TIME FRAME STRUCTURE

In the dedicated physical channel (DPCH/DCH), which contains the user information data, duplexing of the passband signal is accomplished using TDD. The main unit for TD-SCDMA using TDD is a 10ms radio frame, which is divided into two 5ms sub-frames. These sub-frames are further subdivided into seven time slots, of which at least two are reserved for uplink and downlink transmissions and the other five can be either. Figure 2.8 illustrates an example of the most basic unit of TD-SCDMA.

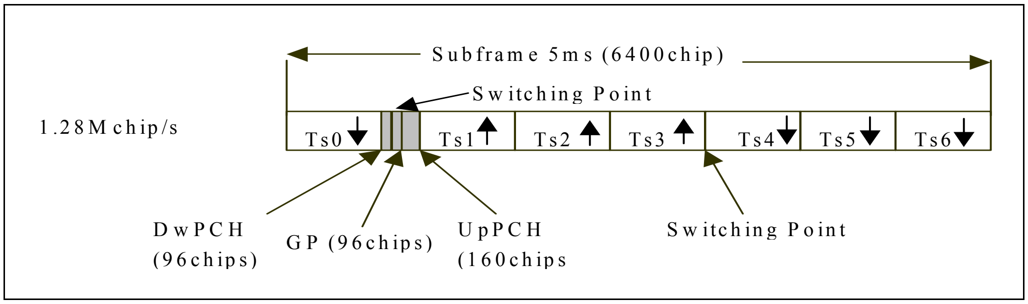

Figure 2.8. This is an example sub-frame for TDD (low chip rate option) with four downlink and three uplink time slots. From Ref. [10].

In this structure, the first time slot (Ts0) is dedicated for downlink and Ts1 is dedicated for uplink. In addition to the user data, two pilot channels and a small guard period are inserted at the switching point between the dedicated downlink and uplink time slots. The remaining five time slots (Ts2-Ts6) can be used for either uplink or downlink transmissions based on user demand.

In each sub-frame, the downlink pilot channel (DwPCH) and uplink pilot channel (UpPCH) are used to maintain synchronization and power control between the user and base station. By calculating the actual time difference between the transmitted downlink synchronization burst and the received uplink synchronization burst, the base station can estimate the propagation delay between itself and the user. This measurement can then be used to calculate the number of synchronization shift (SS) symbols that, when transmitted on the next available downlink time slot, will help maintain uplink synchronization. If re-synchronization is needed, on the next available downlink time slot the base station instructs the user to shift the data transmission by 1/8 chips or any multiple thereof. Since the orthogonality of the signal relies upon signal synchronization, without the DwPCH and UpPCH there would most certainly be interference between users on the same channel. Figure 2.9 illustrates the location of the SS symbol along with the transmitter power control (TPC) symbol. The user uses the TPC to instruct the base station to increase or decrease the transmitter power level as needed to reduce multi-user interference.

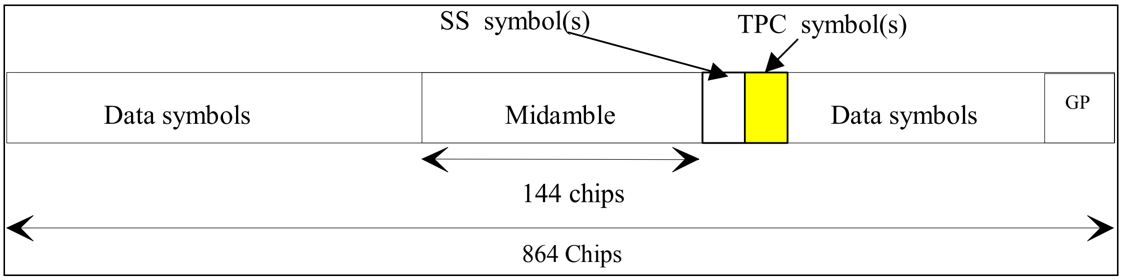

Figure 2.9. The location of the synchronization shift (SS) and transmitter power control (TPC) symbols within one time slot. From Ref. [11].

Each time slot, whether downlink or uplink, is 675µs long. A standard time slot is illustrated in Figure 2.10. In all cases, portions of the encoded and spread information data is contained within two 352 chip time blocks, separated by a 144 chip midamble, and followed by a 16 chip guard period. The purpose of the midamble block is to provide training sequences, which allow the base station to estimate the channel impulse response of all active users in a cell [11] and the user to identify an assigned channel. Each user within a given cell has a time-shifted version of the same midamble code, and each cell is assigned a different midamble code. By correlating the received cyclic sequence with a known reference, the radio frequency (RF) channel impulse response can be estimated. The base station receiver can then use this information to accommodate for fading channels.

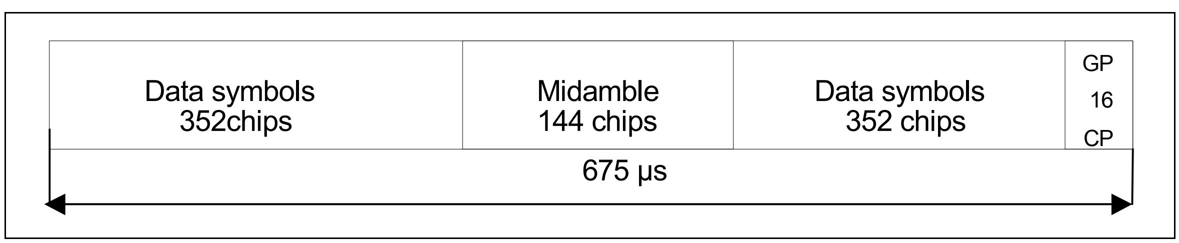

Figure 2.10. Burst structure for normal traffic time slot. From Ref. [10].

To separate multiple users on the same channel, TS-SCDMA employs CDMA using OVSF as described in the previously. This allows up to sixteen users per physical channel, which can be varied depending on the requested user data rates. Figure 2.11 illustrates one sub-frame and how up to 16 users (codes) can be transmitted on the same frequency band (1.6 MHz bandwidth).

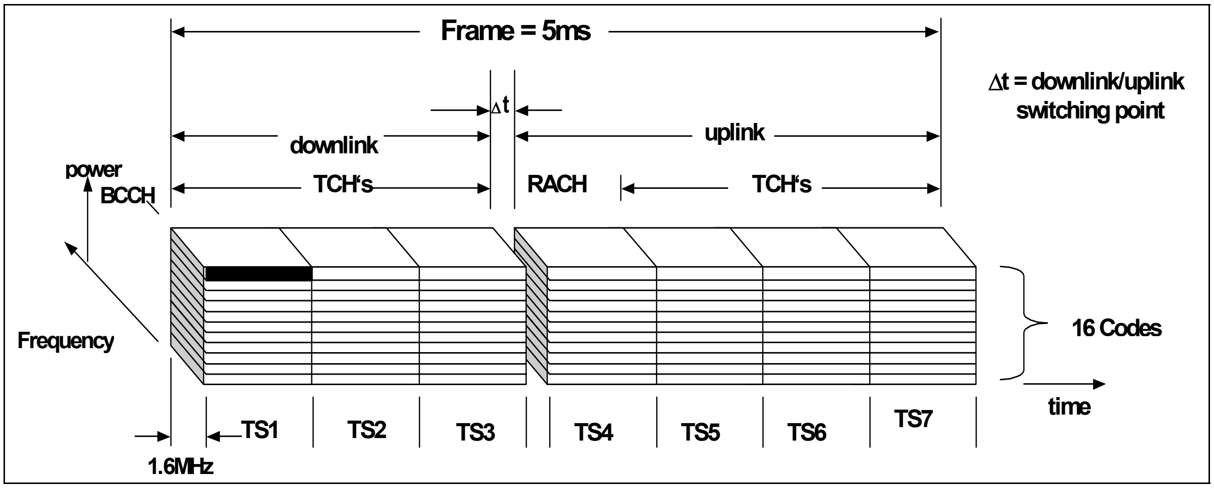

Figure 2.11. TDMA/TDD subframe for symmetric CDMA multi-user transmission of TD-SCDMA. The shaded block is one resource unit (RU). From Ref. [12].

In TD-SCDMA, each user can be assigned one or more OVSF codes depending on the requested user data rate and the number of users on each physical channel. We define each OVSF code of SF=16 on a given time slot as a resource unit (shown in Figure 2.11 as the shaded block), and each user can have access to multiple RUs. In this manner, if a user was assigned two OVSF codes instead of one they would still have access to seven time slots, but instead of seven they now have fourteen available RU's. By using packet data, this allows a user to transmit more symbols in a given unit of time to achieve higher information data rates. Table 2.2 lists the uplink and downlink reference measurement channel data rates and spreading factors used in TD-SCDMA.

Table 2.3. Sub-frame resource allocation for various user data rates.

Information Data Rate		12.2 kbps	64 kbps	144 kbps	384 kbps	2048 kbps
Downlink	Spreading Factor	SF=16	SF=16	SF=16	SF=16	SF=1
	OVSF Codes required	2	8	8	10	1
	Time Slots required	1	1	2	4	5
	Resource Units Allocated	2	8	16	40	80
Uplink	Spreading Factor	SF=8	SF=2	SF=2	1 SF=2 1 SF=8	*NA*
	OVSF Codes required	1	1	1	1	*NA*
	Time Slots required	1	1	2	4	*NA*
	Resource Units Allocated	2	8	16	40	*NA*

To allow multiple users on the same physical channel, or allow one user the ability to transmit multiple OVSF codes on the same timeslot, the TD-SCDMA transmitter utilizes a multiplexer as shown in Figure 2.12. In this figure, the values γ are weight factors which vary according to the spreading factor used, and β represents the overall transmit power gain. Because the signals are orthogonal to one another there should be little to no interference between them at the receiver.

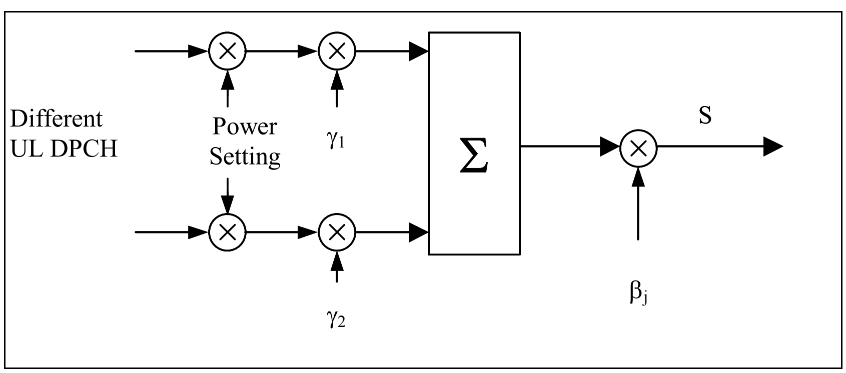

Figure 2.12. Combination of different physical channels in uplink. After Ref. [8].

C. TRANSMISSION AND RECEPTION

As stated before, one of the main advantages of TD-SCDMA is that the transmit and receive frequencies are the same. TD-SCDMA utilizes TDD to duplex both downlink and uplink transmission on the same 1.6 MHz bandwidth carrier.

1. Physical Channels

TD-SCDMA employs two types of physical channels, dedicated physical channels (DPCH) and common physical channels (CPCH). Sections A and B of this chapter dealt mainly with the structure of the DPCH, whereas this section with deal more exclusively with the CPCH. The frame structure of the two channels is identical. The only difference between the two is that the DPCH carries user data information, whereas the CPCH carries control data information.

The CPCH is comprised of several transport channels, which includes but is not limited to, the broadcast channel (BCH), forward access channel (FACH), paging channel (PCH), random access channel (RACH), uplink shared channel (USCH), downlink shared channel (DSCH), and the high speed downlink shared channel (HS-DSCH). Many of these channels are formatted with FEC coding and use the same spreading technique as the DPCH. Because TD-SCDMA does not dedicate a separate 1.6 MHz frequency band for the CPCH, the control data is intermixed with the DPCH during specific time slots and OVSF codes. For example, the dedicated BCH is mapped onto the Primary Common Control Physical Channel (P-CCPCH) and is always transmitted on Ts0, the first dedicated downlink timeslot, using channelization codes $c_{Q=16}^{(k=1)}$ and $c_{Q=16}^{(k=2)}$. The BCH contains the location of all other common transport channels, which can be intermixed throughout the radio frame on other RU's. Figure 2.13 illustrates the location of the P-CCPCH.

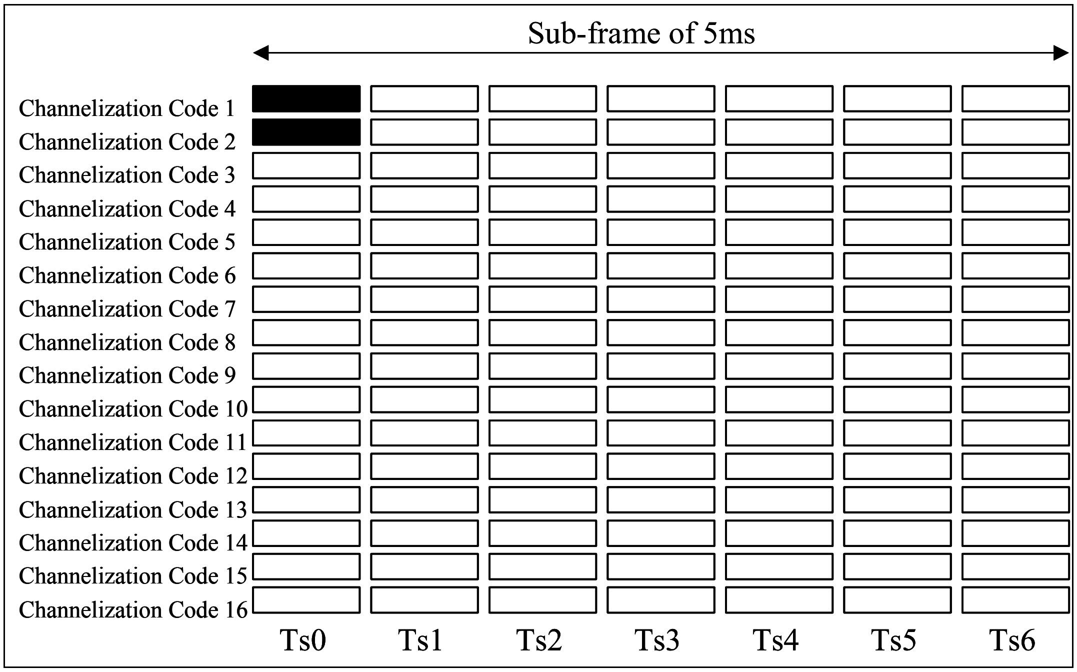

Figure 2.13. Location of the Primary Common Control Physical Channels (P-CCPCH 1 and 2) on the actual physical channel. The P-CCPCH (shaded regions) are always on Ts0 using channelization codes $c_{Q=16}^{(k=1)}$ and $c_{Q=16}^{(k=2)}$.

2. Receiver Characteristics

Of all the technical specifications illustrated and explained in the standard, there is no reference as to how the TD-SCDMA receiver is physically designed. There are of course detailed descriptions as to the minimum reception requirements, but there are no instructions on how to implement them.

As shown previously in Figure 2.4, the transmitter consists of an IQ modulator and two root-raised cosine pulse-shaping filters. To design an appropriate receiver, the transmitter was reverse engineered and implemented in reverse order. To begin, since the transmitter utilizes an IQ modulator, an identical IQ demodulator was placed at the receiver. This type of demodulation creates a baseband reproduction of the original signal and another at twice the carrier frequency (ω_c). To remove the high-frequency component, matching finite impulse response (FIR) lowpass filters are required. By looking at the frequency response of the raised cosine filter in Figure 2.5, we see that this is a lowpass filter with exactly the desired bandwidth and cutoff frequency. Using root-

raised cosine filters matched to the ones in the transmitter, we can accomplish both lowpass filtering and satisfy the Nyquist criterion for reducing ISI.

The next step is to sample the time domain output of the FIR filters at the chiprate and pass the resulting digital waveform through a CDMA receiver. If the synchronized received signal has the same scrambling and OVSF code as the one used being used by the receiver, the original QPSK or 8PSK complex data sequence will be extracted. If the received signal is out of synchronization or is scrambled and spread using different codes, the receiver will not recreate the original data.

To fully reproduce the original information data as sent by the base station or user, the receiver must demodulate the complex data sequence and un-encode the resulting digital data. Depending on the number of users on a physical channel and the requested data rates, this process could involve interleaving, puncturing, and turbo or convolutional decoding. Figure 2.14 illustrates the complete theoretical receiver as designed by the author.

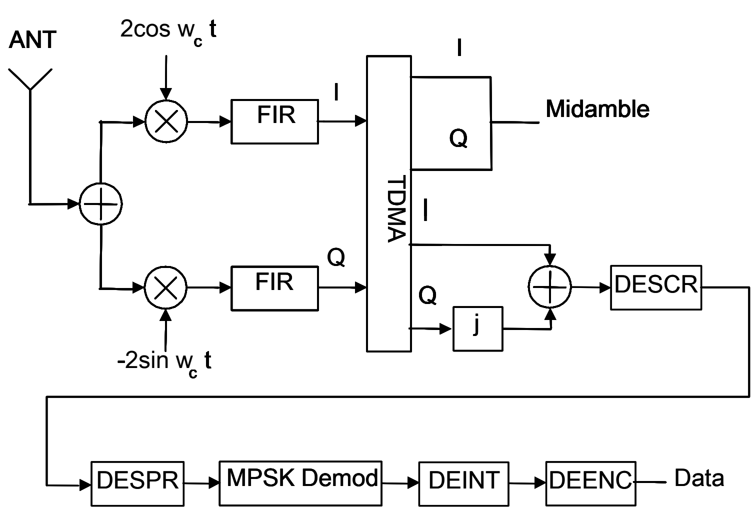

Figure 2.14. Possible TD-SCDMA receiver for DPCH. After the I/Q summation, the signal is descrambled (DESCR), despread (DESPR), demodulated, de-interleaved (DEINT), and de-encoded (DEENC). After Ref. [13].

D. COMPARISON WITH OTHER 3G PROPOSALS

Although this thesis deals mainly with TD-SCDMA, this is but one of many 3G cellular communications standards. Figures 1.1 and 1.2 show most of the 3G proposals that are currently in development. From Figure 1.2 we can see that the two main competitors for the 3G market are wideband code-division multiple access (WCDMA) and CDMA2000, which have solid core networks (GSM and IS-95, respectively) existing throughout the world. In this section the author will compare TD-SCDMA to these two systems and attempt to draw conclusions as to which system is better.

1. WCDMA

WCDMA, like TD-SCDMA, is based on the GSM core network that is prevalent throughout Europe and parts of Asia and North America. This makes the new technology very attractive to existing owners of this core network since it will be cheaper to upgrade to WCDMA or TD-SCDMA than it will be to convert to CDMA2000. The main drawback of WCDMA is that even though there are provisions for backward compatibility with existing GSM systems, the new technology requires an equipment change-out to use the new handsets. Eventually, the entire GSM network will be phased out and replaced with WCDMA/UMTS.

WCDMA is being marketed as the preferred upgrade path from GSM to 3G. With a bandwidth of 5MHz, one WCDMA channel (uplink or downlink) can be "sandwiched" into the current GSM frequency spectrum while still providing GSM service. Since each GSM channel has a bandwidth of 200 kHz, WCDMA combines 25 adjacent GSM channels to obtain one WCDMA channel. TD-SCDMA can also use this technique to utilize the current GSM spectrum, but requires only 8 adjacent GSM channels instead of 25.

Another important feature of WCDMA is that this new technology uses code-division multiple access in a manner similar to TD-SCDMA. Both employ orthogonal variable spreading factors (OVSF) to allow multiple users on the same physical channel, which is a significant improvement over GSM. However, WCDMA has the ability to support user data rates of 2048 kbps with a spreading factor of $Q_k = 4$. In comparison, TD-SCDMA can only support user data rates that high if no spreading is employed

($Q_k = 1$). This means that WCDMA can theoretically support up to four users at 2048 kbps on the same physical channel while TD-SCDMA can only support one. Even so, WCDMA occupies six times the bandwidth of TD-SCDMA, and using more TD-SCDMA channels allows TD-SCDMA to support just as many users as WCDMA at that data rate.

If we compare technology, WCDMA like TD-SCDMA will support data rates up to 2 Mbps, but unlike TD-SCDMA this system uses a chipping rate of 3.84 Mcps, both TDMA and FDD, and has a total passband bandwidth of 10 MHz (5 MHz per channel). Neglecting overhead, transmit diversity, the use of smart antennas, or sectoring, we find that given a 10 MHz spectrum TD-SCDMA can support a total of 96 voice users at 12.2 kbps while WCDMA can support 64 voice users at 12.2 kbps (WCDMA can increase this to 98 users at 7.95 kbps using a speech codec) [15]. If we add 120° sectoring, TD-SCDMA can support 288 users/cell, while WCDMA can support 192 users/cell at 12.2 kbps or 294 users/cell at 7.95 kbps. Keep in mind that these calculations for the total number of possible users on the downlink were computed neglecting any overhead for transport channels.

For modulation, in addition to QPSK WCDMA employs hybrid phase-shift keying (HPSK) on the uplink channel and uses 10 ms radio frames much like TD-SCDMA. Even though each user in WCDMA has access to all timeslots for a given code, the user can use different spreading factors for each timeslot. This means that in one timeslot the user could transmit a data rate of 215 kbps and the next slot transmit at only 45 kbps. Because the chipping rate does not change, WCDMA like TD-SCDMA maintains a constant bandwidth of 5 MHz regardless of the user data rate. A major difference between the two standards is that at 2.048 Mbps WCDMA employs spreading and rate ½ FEC coding. In comparison, TD-SCDMA uses 8PSK, no spreading, and no FEC coding to achieve this data rate. This means that WCDMA at higher user data rates will have a lower probability of bit error rate and be harder to intercept than TD-SCDMA.

Even though it appears that WCDMA is comparable to TD-SCDMA in most aspects, the main advantage of TD-SCDMA over WCDMA is that TD-SCDMA uses

TDD and therefore requires less bandwidth. WCDMA requires a paired spectrum for a total of 10MHz to obtain uplink and downlink transmissions, whereas TD-SCDMA requires only one spectrum of 1.6 MHz. This means that WCDMA requires more spectrum planning and larger portions of that spectrum to provide even just one forward and reverse channel. In the unpaired spectrum bands, TD-SCDMA can be placed between and around existing bands used for other services. This means that if there exists a 3.2 MHz gap in the current allocated frequency spectrum, the service providers for TD-SCDMA could utilize that portion for two TD-SCDMA physical channels whereas not even one WCDMA channel could be employed.

2. CDMA2000 1x and 1xEV

CDMA2000 1x and 1xEV, hereafter jointly referred to as CDMA2000, is an extension of the current IS-95 technology that is in widespread use in the Americas. Like the older system and WCDMA, CDMA2000 uses FDD to separate the forward and reverse links for uplink and downlink communications. This technique is entirely different from TD-SCDMA, which uses TDD to place the forward and reverse link on the same frequency. CDMA2000 1x, the first stage of CDMA2000, will support user data rates up to 153 kbps and CDMA2000 1xEV will support rates up to 307.2 kbps. Later implementations will support data rates of 625 kbps and 2.4 Mbps.

To keep CDMA2000 compatible with IS-95, which uses a chip rate of 1.2288 Mcps and a bandwidth of 1.25 MHz, CDMA2000 employs multi-carrier modulation. This means that in order to achieve data rates up to 307.2 kbps, CDMA2000 1xEV spreads the information data using a chip rate of 1.2288 Mcps and employs three different orthogonal frequency carriers to transmit the data. Therefore, CDMA2000 1xEV effectively has a chipping rate of 3.6864 Mcps and bandwidth of 3.75 MHz even though the actually chipping rate/channel is 1.2288 Mcps and the processing gain remains unchanged. Currently, CDMA2000 1x has twice the user capacity of current IS-95 systems and CDMA2000 1xEV will have three times the capacity. The biggest advantage of this is that new handsets are completely compatible with the older base stations, and to utilize the faster data rates only requires a software revision and card change at the base station.

If we compare the number of users per channel for the two systems, we see that TD-SCDMA falls short of the competition. To make a fair comparison we will choose a spectrum bandwidth of 5 MHz since this fits nicely with two CDMA2000 channels (two forward and two reverse channels) and three TD-SCDMA channels. Because CDMA technology normally employs sectoring, we will include 120˚ sectoring in this comparison. As such, CDMA2000 1x can support 192 users/cell at 9.6 kbps on the reverse link while TD-SCDMA can only support 144 voice users at 12.2 kbps. The author believes the reason for this disparity is that TD-SCDMA has a maximum spreading factor of 16 whereas CDMA2000 uses 128 Walsh spreading codes.

In this instance we see that TD-SCDMA does not compare well with CDMA2000. Although TD-SCDMA utilizes a smaller bandwidth, 1.6 MHz is not that much smaller than 2.5 MHz. Also, new TD-SCDMA base stations do not provide compatibility for older GSM handsets, whereas CDMA2000 does for older IS-95 systems. This means that while GSM technology will be completely phased out, IS-95B handsets will still be compatible with the newer base stations.

3. Which is Better?

Despite any benefits that either system has over the other, the author does not foresee any major service provider of GSM switching to CDMA2000 or an IS-95 provider switching to TD-SCDMA or WCDMA. The cost of installing or replacing an entire base station is too much considering that most base stations are relatively new (built within the past decade). More than likely, the service providers will chose a 3G cellular system that provides some backward compatibility for their current customers and equipment. This means that most of the world has already chosen WCDMA as their 3G upgrade, but CDMA2000 is a very attractive option for those companies whom have invested much in CDMA technology or are looking to install a fresh system.

Although TD-SCDMA does have an advantage in using one physical channel for both uplink and downlink transmissions in a narrow bandwidth, this is outweighed by the fact that both WCDMA and CDMA2000 can handle more users/channel with FEC coding. Unless China foregoes WCDMA in favor of their homegrown TD-SCDMA, the author does not foresee TD-SCDMA gaining a large share of the 3G market.

THIS PAGE INTENTIONALLY LEFT BLANK

III. ANALYSIS OF TD-SCDMA

A. PERFORMANCE ANALYSIS OF TD-SCDMA

1. TD-SCDMA Signals in the Presence of AWGN

To evaluate the performance of TD-SCDMA under adverse conditions we will begin with the most basic of noise channels, the additive white Gaussian noise (AWGN) channel. AWGN is defined as a zero-mean Gaussian random process whose power spectral density (PSD) is flat or *white* over all frequencies. The source of AWGN is thermal noise, which "is caused by the thermal motion of electrons in all dissipative components" [9]. Since the systems that are adversely affected by AWGN rely upon electrical conduction to operate, there are few ways to minimize the effect of AWGN.

To begin the evaluation, we will need a mathematical equation to represent the received signal at the input to the receiver. Assuming perfect synchronization and using (2.4) with Figure 2.4, we can interpolate that the total received signal $s(t)$ is

$$s(t) = \sqrt{2}A_c\left[\mathrm{Re}\left\{d^{(k,i)}(t)\right\}\cos\left(\omega_c t\right) - \mathrm{Im}\left\{d^{(k,i)}(t)\right\}\sin\left(\omega_c t\right)\right] + n(t) \tag{3.1}$$

where $d^{(k,i)}$ is the spread and scrambled data, $\sqrt{2}A_c$ is the received signal amplitude, and $n(t)$ is AWGN with PSD of $N_0/2$. We further define

$$d^{(k,i)}(t) = d_n^{(k,i)}c(t)Cr_0(t) \tag{3.2}$$

where $d_n^{(k,i)}$ is the original complex data to be transmitted, $c(t)$ is the complex spreading and scrambling code, and $Cr_0(t)$ is the root-raised cosine filter impulse response. This will be useful later on in the receiver.

If we assume a receiver design as shown in Figure 2.14, we see that the signal is passed through an IQ demodulator, low-pass filtered with matching root-raised cosine filters, and recombined into a complex expression. We will also assume the filter removes components at twice the carrier frequency, and therefore before despreading we have a

$$r(n) = r_I(n) + j \bullet r_Q(n) \tag{3.3}$$

where

$$r_I(n) = \sqrt{2}A_c \operatorname{Re}\left[d_n^{(k,i)}c(t)\right] + \left\{\left[2n(t)\cos\left(\omega_c t\right)\right] \otimes Cr(t)\right\}\Big|_{t=nT_c} \tag{3.4}$$

and

$$r_Q(n) = \sqrt{2}A_c \operatorname{Im}\left[d_n^{(k,i)}c(t)\right] - \left\{\left[2n(t)\sin\left(\omega_c t\right)\right] \otimes Cr(t)\right\}\Big|_{t=nT_c} \tag{3.5}$$

In both equations, T_c is the chip period and the symbol $\otimes$ represents a convolution of the noise with the root-raised cosine filter impulse response. A point of interest is that we are no longer strictly in the continuous time domain. This is a result of the RRC filter, which samples and holds the output every T_c seconds in order to reduce ISI.

The next section of the receiver takes the sampled signal and reverses the spreading and scrambling procedures that were done in the transmitter. Because TD-SCDMA uses a complex scrambling sequence, the receiver uses a complex conjugate of the spreading and scrambling codes to return the information data. If the receiver had used an exact replica of the transmitter's spreading and scrambling codes, some descrambled chips would undergo a 180° phase shift. For example, if the information data symbol was $1-j$ and the scrambling code was j we would transmit $j \bullet (1-j) = 1+j$. At the receiver, if we used the same scrambling code j we would recover $j \bullet (1+j) = -1+j$, which is the inverse of what we sent. By taking the complex conjugate of the transmitter code, we would recover $-j \bullet (1+j) = 1-j$, which is an exact replica of the original signal.

Using the above, we find that the received signal at the output of the descrambling and despreading stage is

$$\begin{aligned} x(t) = {} & \sqrt{2}A_c c^*(t_c)\left\{d_n^{(k,i)}c(t)\right\} + \\ & c^*(t_c)\left\{\left[2n(t)\cos(\omega_c t) - j2n(t)\sin(\omega_c t)\right] \otimes Cr(t)\right\}\Big|_{nT_c} \end{aligned} \tag{3.6}$$

where $c^*(t)$ is the complex conjugate of the original spreading and scrambling sequence. We can further simplify the first term by using (3.2) to reduce it to $\sqrt{2}A_c d_n^{(k,i)}$ since

$$c^*(t) d_n^{(k,i)} c(t) = d_n^{(k,i)}, \tag{3.7}$$

where the product of $c(t)$ and its complex conjugate $c^*(t)$ is 1. This then leaves us with a replica of the original signal plus a noise term.

To compute the probability of a symbol error we will next evaluate the result of the noise term and see what impact this has on the total received signal.

Looking just at the contribution of noise to the received signal, we will assign $n'(nT_c)$ the value of

$$n'(nT_c) = c^*(nT_c)\left\{\left[2n(t)\cos(\omega_c t) - j2n(t)\sin(\omega_c t)\right] \otimes Cr(t)\right\}\Big|_{nT_c}. \tag{3.8}$$

Using the theorems for the PSD of a heterodyned Gaussian white noise process and the PSD for a random process passed through a linear system, we conclude that the PSD of the in-phase and quadrature noise before sampling is

$$S_{n_{I,Q}}(f) = N_0 \left|Cr(f)\right|^2, \tag{3.9}$$

where $Cr(f)$ is the transfer function of the root-raised cosine filter.

Thus, the product of the heterodyned noise terms with the root-raised cosine filter produces band-limited Gaussian noise with the power spectral density

$$S_n(f) = \begin{cases} N_0 & 0 \le |f| \le \dfrac{(1-\alpha)}{2T_c} \\ \dfrac{N_0}{4}\left[1+\cos\left[\dfrac{\pi(|f|\bullet 2T_s - 1 + \alpha)}{2\alpha}\right]\right]^2 & \dfrac{(1-\alpha)}{2T_c} \le |f| \le \dfrac{(1+\alpha)}{2T_c} \\ 0 & |f| \le \dfrac{(1+\alpha)}{2T_c} \end{cases} \tag{3.10}$$

Because the RRC filter also samples the incoming signal every T_c seconds, we need to look at the effect of sampling the band-limited Gaussian noise. From [9] we have the equation for sampling and holding a continuous time function

$$X_s(f) = P(f)\frac{1}{T_s}\sum_{n=-\infty}^{\infty} X(f - nf_s). \tag{3.11}$$

Here $X_s(f)$ is the discrete time Fourier transform of the sampled signal, $X(f)$ is the continuous time Fourier transform to be sampled, $P(f) = T_s \operatorname{sinc}(f T_s)$ is the holding function, and $f_s = 1/T_s$ is the sampling frequency. Since the RRC filter is designed to sample at a rate equal to the chipping rate we see that the process violates the Nyquist criterion of $f_s \geq 2f_m$, where f_m is twice the baseband bandwidth of the signal to be sampled. By violating Nyquist's criterion, the sampling procedure will introduce aliasing at the output of the RRC filter. In this case the sampling frequency is equal to the bandwidth and, before being multiplied by $P(f)$, causes the filter to recreate spectrally flat noise for all frequencies. Therefore, the power spectral density output of the RRC filter is $T_c \operatorname{sinc}^2(f T_c) N_0 R_c$.

Before we can proceed to the next stage, we must combine the two filtered noise terms into one complex sequence. Because the noise PSD is identical at the output of both RRC filters, this means that the total noise power is twice the PSD of one filter, or $2T_c \operatorname{sinc}^2(f T_c) N_0 R_c$.

After the signal is combined into a complex waveform, the receiver requires that we despread and descramble using the complex chipping sequence $c^*(nT_c)$. Analogous to [7], we know that the complex valued chipping sequence consists of a polar random binary wave periodic over T_c seconds. The PSD of this particular polar random wave is given by

$$S_{c^*}(f) = T_c \operatorname{sinc}^2(f T_c). \tag{3.12}$$

This means that the PSD after chipping is the convolution of noise PSD with the chipping sequence and has the form

$$S_{n'}(f) = T_c \mathrm{sinc}^2\left(f\,T_c\right) \otimes 2T_c \mathrm{sinc}^2\left(f\,T_c\right) N_0 R_c\,. \qquad (3.13)$$

This convolution is not intuitively easy; therefore we will take the inverse Fourier transform and multiply in the autocorrelation domain. The autocorrelation of $T_c \mathrm{sinc}^2\left(f\,T_c\right)$ is

$$R_x\left(\tau\right) = \begin{cases} 1 - \dfrac{|\tau|}{T_c} & |\tau| \le T_c \\ 0 & otherwise \end{cases} \qquad (3.14)$$

and thus the multiplication of the two is

$$R_n\left(\tau\right) = \begin{cases} 2N_0 R_c \left(1 - \dfrac{|\tau|}{T_c}\right)^2 & |\tau| \le T_c \\ 0 & otherwise \end{cases}. \qquad (3.15)$$

Taking the Fourier transform to convert back into the power spectral density domain, we get

$$S_n = 2N_0 R_c \left\{ \frac{T_c}{2} \mathrm{sinc}^2\left(f\,\frac{T_c}{2}\right) - \left(\frac{1}{T_c}\right)^2 \left(\frac{1}{2\pi}\right)^2 \frac{\partial^2}{\partial f^2}\left[T_c \mathrm{sinc}\left(f\,T_c\right)\right] \right\}. \qquad (3.16)$$

The final stage of the receiver before converting the QPSK or 8PSK symbols back into binary ones and zeroes is either an integrator or summer. The integrator or summer combines consecutive chips over one bit interval to reconstruct the information data. In this analysis we will choose an integrator and use the theorem that $S_{out}(f) = S_{in}(f)\left|H\left(f\right)\right|^2$, where $H\left(f\right) = \mathrm{sinc}(f\,T_b)$ is the transfer function of the integrator. This means that the noise PSD at the output of the receiver is

$$S_{n_{out}} = 2N_0 R_c \left\{ \frac{T_c}{2} \mathrm{sinc}^2\left(f\,\frac{T_c}{2}\right) - \left(\frac{1}{T_c}\right)^2 \left(\frac{1}{2\pi}\right)^2 \frac{\partial^2}{\partial f^2}\left[T_c \mathrm{sinc}\left(f\,T_c\right)\right] \right\} \left\{\mathrm{sinc}^2\left(f\,T_b\right)\right\}. \qquad (3.17)$$

Since we are really only interested in the noise power at the output of the receiver we need to compute

$$\sigma^2 = \int_{-\infty}^{\infty} S_{n_{out}}\left(f\right)df \ . \tag{3.18}$$

When we substitute (3.17) in (3.18) and integrate, we find that the total power is approximately $\frac{N_0 R_c}{Q_k} = \frac{N_0}{T_b}$ for $Q_k \in [1,2,4,8,16]$. This means that the chipping sequence has minimal effect on the Gaussian noise. When we compare this to a generic QPSK direct sequence spread spectrum receiver without a low pass filter at the input, we see that the TD-SCDMA receiver has exactly the same noise power and similar results.

Now that we know the noise power of our signal we will compute the phase of our complex valued signal and use a threshold detector to determine the corresponding MPSK symbol. If the computed phase lies between two threshold lines, an estimation is made as to what the original MPSK symbol was. In the simulations we take the difference between this estimate and the actual transmitted symbol to determine the actual probability of symbol error.

From [15] we conclude that the probability of a symbol error for $M \geq 8$ can be approximated by

$$P_M \approx 2Q\left(\sqrt{2 \cdot \log_2\left(M\right)\frac{E_b}{N_0}}\sin\left(\frac{\pi}{M}\right)\right). \tag{3.19}$$

Because this equation is not a good approximation for M=2 or M=4, for QPSK we will use the exact expression of

$$P_4 = 2Q\left(\sqrt{\frac{2E_b}{N_0}}\right)\left[1 - \frac{1}{2}Q\left(\sqrt{\frac{2E_b}{N_0}}\right)\right] \tag{3.20}$$

where E_b / N_0 is the signal to noise ratio at the input to the receiver.

In the computer simulation, we will leave the signals at baseband to analyze a TD-SCDMA signal being transmitted and received. This procedure is identical to analyzing the system at passband as long as we take into account the effect that heterodyning and filtering has on the signal and noise terms. When we run the simulation

for the probability of symbol error for this receiver in the presence of AWGN, we obtain results as shown in Figure 3.1.

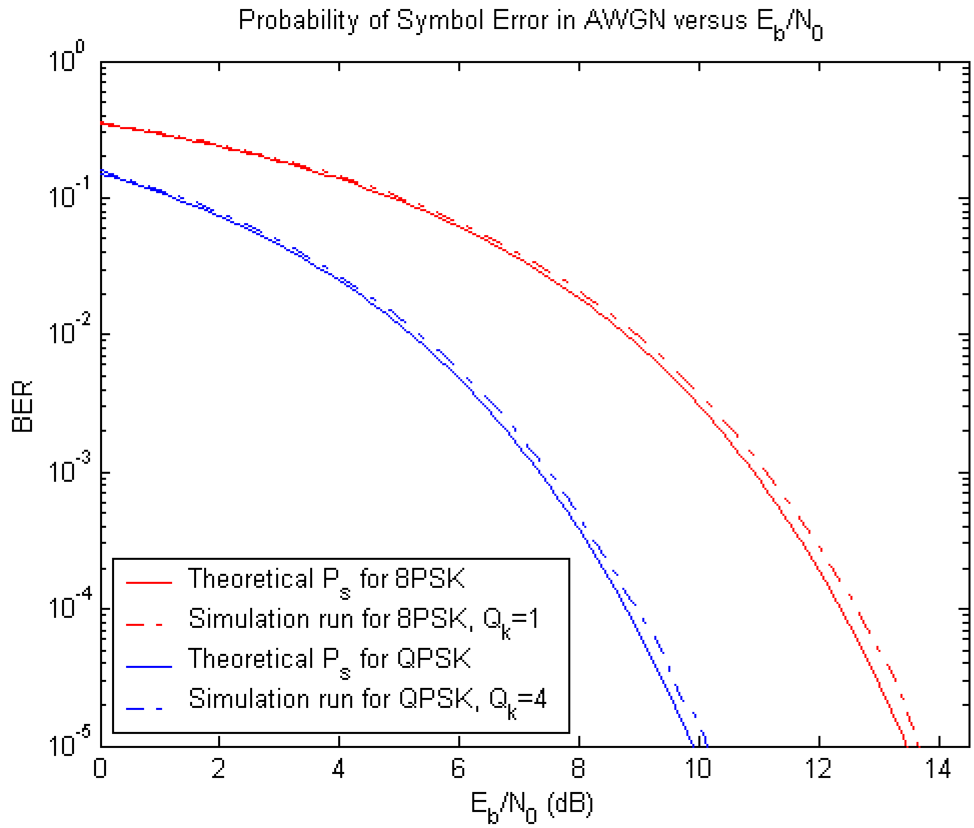

Figure 3.1. Probability of symbol error versus E_b/N_0 for TD-SCDMA using QPSK and 8PSK in AWGN.

From these plots we see that the simulation results match up very well with the theoretical solutions. The only discrepancy is that the simulation results are approximately 0.1 dB greater than the theoretical solution for large E_b / N_0. This is due in part to the fact that the power of the noise is not exactly N_0/T_b but is a close approximation. The other reason is that the author was not able to exactly match E_b/N_0 at the input to the receiver. Since the power of the transmitted signal is spread over six chip periods and overlaps with consecutive bits, this makes an exact E_b/N_0 difficult to approximate at the input to the receiver. Otherwise, these simulations results confirm that the computer code is written and functioning correctly. The next step is to introduce multipath fading.

2. In AWGN with Rayleigh Fading

An analysis of the performance of TD-SCDMA in the presence of AWGN is important, but the performance of our system in the presence of fading is even more so. In the previous section we discussed errors introduced by thermal noise in the electronic components themselves. In this section we will discuss the effects of transmitting and receiving a signal over a fading channel.

Signal fading is defined as amplitude variations in a received signal due to a time-varying multipath channel [15]. When a signal is transmitted over the air, the electrical signal interacts with ions in the atmosphere, which cause some of the signal to become dispersed and scatter in other directions. Other contributors to signal fading are objects that lie between the transmitter and receiver. Reflections of the transmitted signal off of stationary objects or moving vehicles can arrive at a receiver to either add constructively or destructively. Since the movement of ions is haphazard, and there are any number of objects that lie between the transmitter and receiver, we can approximate the fading channel as a random process that affects our signal. In this section we will discuss the most basic of fading channels, the Rayleigh fading channel.

A Rayleigh fading channel occurs when we consider that there are no stationary objects between the transmitter and receiver. This is an unlikely scenario, but the analysis is much simpler than assuming multiple non-moving objects. If we assume all our variables as being in motion we can model our environment as a zero-mean, complex valued Gaussian process whose envelope is Rayleigh distributed [15]. This means that the amplitude of our received signal at any given point in time has a Rayleigh probability distribution defined as

$$f_R(r) = \frac{r}{\sigma^2} e^{-r^2/2\sigma^2} \qquad r \geq 0 \quad \sigma > 0. \tag{3.21}$$

The mean of this distribution is $\overline{R} = \sigma\sqrt{\pi/2}$ and the variance is $\sigma_R^2 = (1-\pi/4)2\sigma^2$. We can also define the average power of the signal

$$\overline{r^2} = \sigma_R^2 - \left(\overline{R}\right)^2 = 2\sigma^2\left(1-\frac{\pi}{4}\right) - \sigma^2\frac{\pi}{4} = 2\sigma^2. \tag{3.22}$$

To make calculations for the probability of symbol error easier we will assume that the channel frequency response is constant over the signal bandwidth and the amplitude of the signal does not vary over one symbol duration. This type of analysis is termed a frequency-nonselective (or flat), slow fading channel. By assuming that the fading is constant over one symbol duration we can also assume that spreading has no effect on the fading. This may not be the case if $T_{symbol} > T_{coherence}$.

The performance of MPSK over frequency-nonselective, slow fading Rayleigh channels is well is documented in [15]. To begin our analysis, we will use the basic equation for the probability of symbol error with Rayleigh fading,

$$P_s = \int_0^\infty P_s(\gamma_s) f_{\Gamma_s}(\gamma_s)\,\delta\gamma_s, \tag{3.23}$$

where $P_s(\gamma_s)$ is the probability of symbol error for an MPSK signal in AWGN, and $f_{\Gamma_s}(\gamma_s)$ is the probability density function of γ_s for the Rayleigh fading channel. Using (3.19) in (3.23), we can evaluate this equation to obtain

$$\begin{aligned} P_s &= \int_0^\infty 2Q\left(\sqrt{2\gamma_s}\sin\frac{\pi}{M}\right)\frac{1}{\overline{\gamma_s}}\exp\left(-\gamma_s/\overline{\gamma_s}\right)\delta\gamma_s \\ &= 1-\sqrt{\frac{\overline{\gamma_s}\sin^2\left(\frac{\pi}{M}\right)}{1+\overline{\gamma_s}\sin^2\left(\frac{\pi}{M}\right)}} \end{aligned} \tag{3.24}$$

where γ_s is the symbol energy-to-noise ratio and $\overline{\gamma_s}$ is the average value of γ_s. Since we have assumed a frequency-nonselective, slow fading channel, we know that the average symbol energy is the average power or $2\sigma^2$. With the magnitude of our transmitted symbol as 1 (an MPSK baseband signal has a magnitude of 1 on the unit circle), this means that our average symbol energy-to-noise ratio after fading is $\overline{\gamma_s} = \dfrac{2\sigma^2}{N_0 T_s}$.

When we add Rayleigh fading to our computer simulation for AWGN we obtain results shown in Figure 3.2. As was the case with only AWGN, the author was not able

to accurately set $\bar{E}_b/N_0$ due to the nature of the RRC filter. This explains why the plots for the Rayleigh fading do not exactly match the theoretical solution. Otherwise, the performance of TD-SCDMA in Rayleigh fading is the same as a generic direct sequence spread spectrum signal.

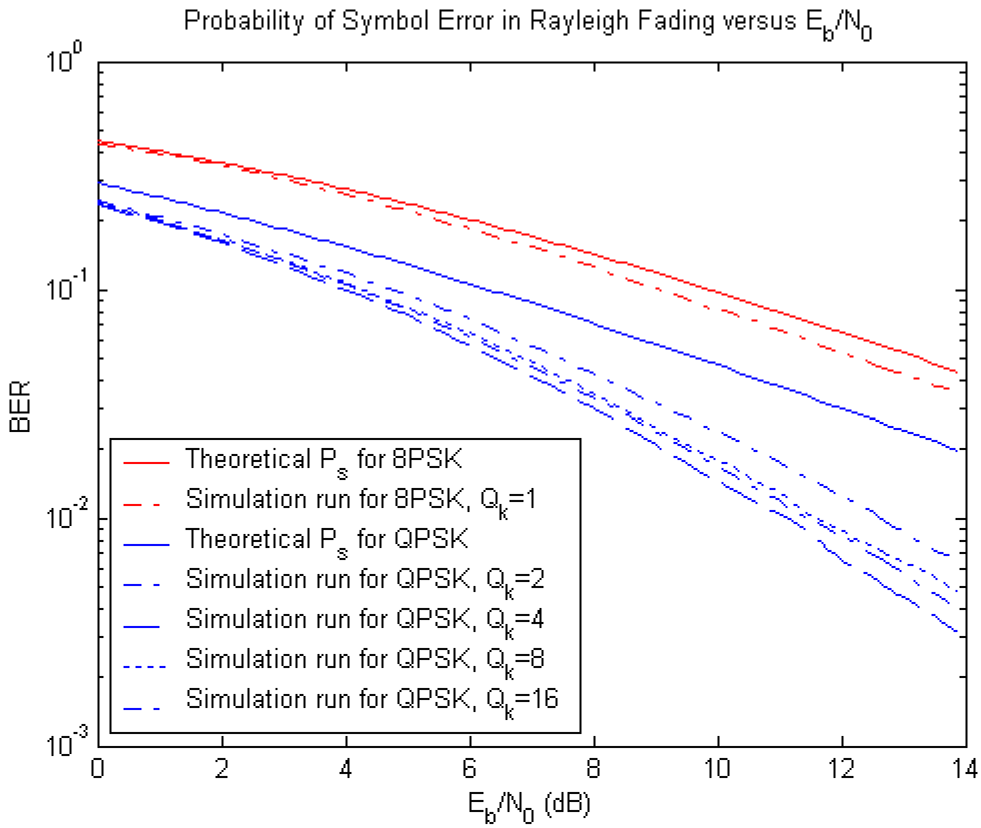

Figure 3.2. Probability of symbol error versus $\bar{E}_b/N_0$ for TD-SCDMA using QPSK and 8PSK in the presence of frequency-nonselective, slow Rayleigh fading.

If we look at the performance of the system without the root-raised cosine filtering we can better match up $\bar{E}_b/N_0$. This yields results that are closer to the theoretical values, and in all aspects we can be consider them to be closer to the actual values than Figure 3.2. The simulation results without the root-raised cosine filter are illustrated in Figure 3.3.

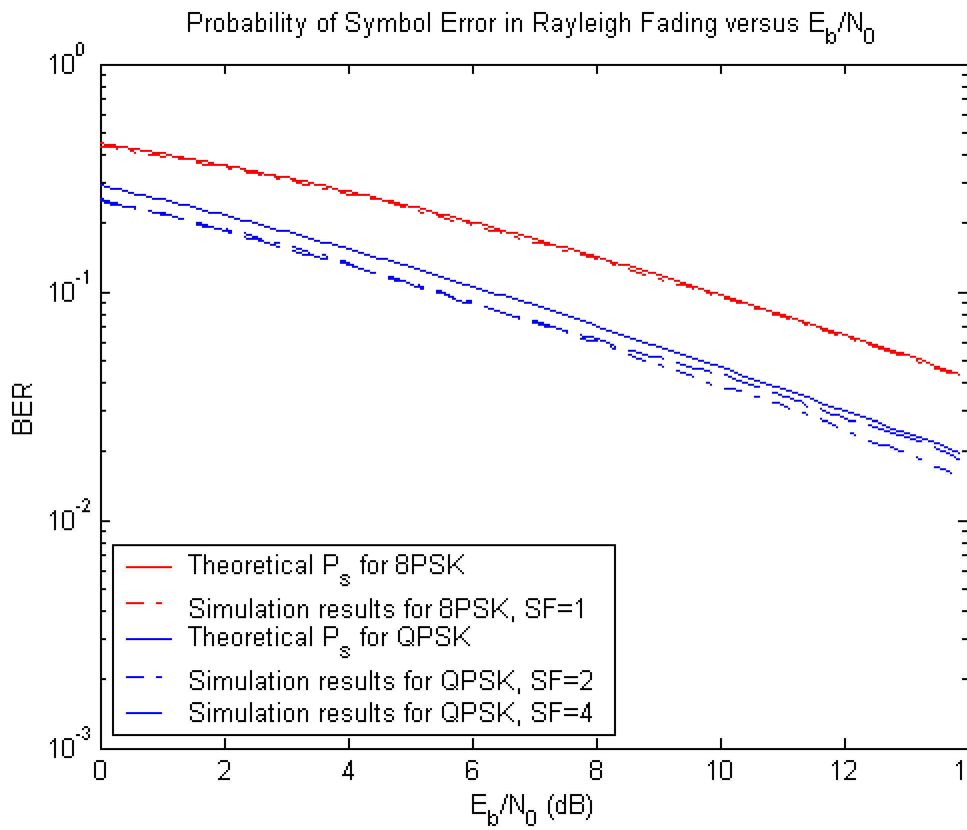

Figure 3.3. Probability of symbol error versus $\bar{E}_b/N_0$ for TD-SCDMA without root-raised cosine filtering using QPSK and 8PSK in the presence of frequency-nonselective, slow Rayleigh fading.

B. PERFORMANCE ANALYSIS WITH JAMMING PRESENT

From a law enforcement and military perspective, from time to time there may be a need to block or hamper a TD-SCDMA transmission. This section will analyze the performance of TD-SCDMA under two adverse conditions that create intentional noise at the receiver. By analyzing these results we can determine which jamming technique is more effective at increasing the probability of error at the mobile or base station receiver.

1. Tone Jamming

One well-documented way of jamming an RF signal is to transmit a sinusoidal wave at the carrier frequency of the desired signal to be jammed. This leads to an additive interference signal of the form

$$s_I(t) = \sqrt{2}A_I \cos\left(\omega_c t + \theta_I\right) \tag{3.25}$$

where θ_I is the phase of the jamming signal and is normally considered a uniform random variable between $[0, 2\pi]$. For this simulation we will assume the worst-case scenario where the tone-jamming signal has exactly the same phase as the carrier frequency of the TD-SCDMA signal. Although the probability of this event happening is exactly zero, this assumption will provide us with an upper bound on the probability of symbol error. Adding the interference signal to the AWGN scenario described previously, we get the results as shown in Figure 3.4.

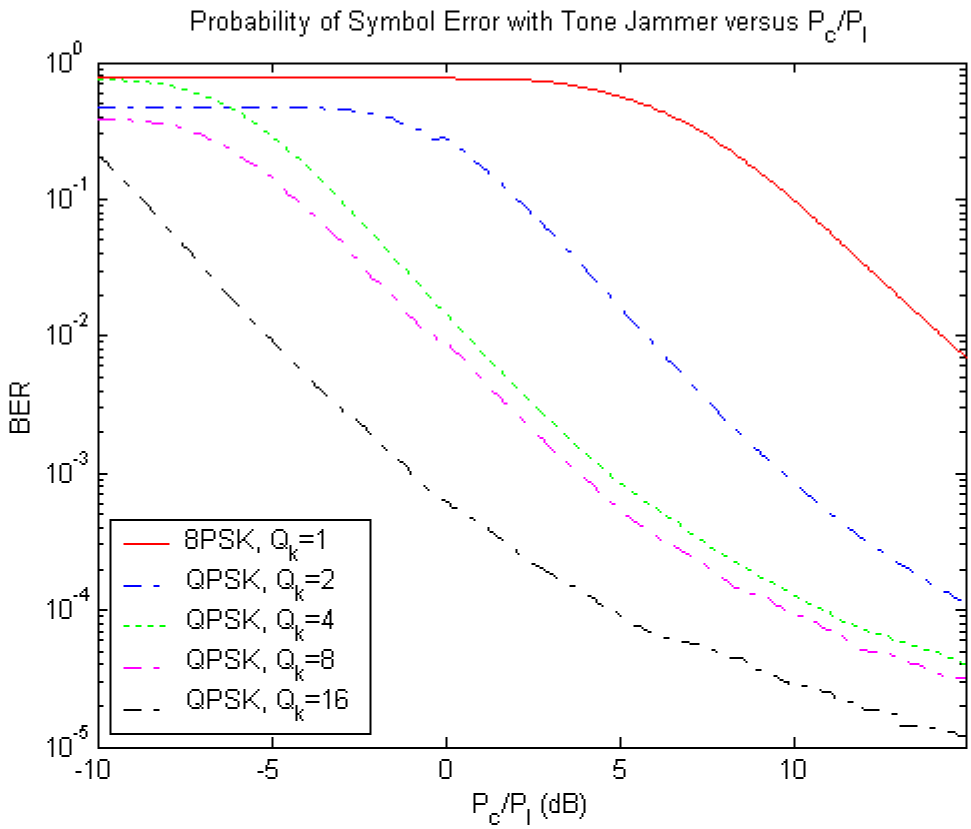

Figure 3.4. Worst case probability of symbol error versus P_c/P_I for a TD-SCDMA signal using QPSK and 8PSK in the presence of tone jamming.

Unfortunately, the author was not able to locate an applicable theoretical solution for a direct sequence spread spectrum MPSK signal with a tone-jamming interferer. The closest theoretical solution found was for a direct sequence spread spectrum QPSK signal with tone jamming when $R_c \gg R_b$ where

$$P_{S,QPSK} = 2Q\left(\sqrt{\frac{2}{\left(\frac{E_b}{N_0}\right)^{-1} + \frac{2P_I}{kP_c}\cos^2(\theta_I)}}\right)\left[1 - \frac{1}{2}Q\left(\sqrt{\frac{2}{\left(\frac{E_b}{N_0}\right)^{-1} + \frac{2P_I}{kP_c}\cos^2(\theta_I)}}\right)\right]. \qquad (3.26)$$

This equation does not produce meaningful results for our TD-SCDMA system since our chipping rate is at times equal to the bit rate, $R_c = R_b$. A more comparable equation was derived by using the probability of bit error for a BPSK signal with a tone jammer

$$P_{b,BPSK} = \frac{1}{2}\left\{Q\left[\sqrt{\frac{2E_b}{N_0}}\left(1 + \sqrt{\frac{P_I}{P_c}}\cos\theta_I\right)\right] + Q\left[\sqrt{\frac{2E_b}{N_0}}\left(1 - \sqrt{\frac{P_I}{P_c}}\cos\theta_I\right)\right]\right\}. \qquad (3.27)$$

In this equation P_I / P_c is the ratio of the jamming interference power to the BPSK signal power and θ_I is the phase of the tone-jamming signal relative to the BPSK carrier frequency phase. By taking into account that for a direct sequence spread spectrum P_I is related to P_c by the spreading factor Q_k and setting $\theta_I = 0$, we can rewrite (3.27) as

$$P_{b,BPSK} = \frac{1}{2}\left\{Q\left[\sqrt{\frac{2E_b}{N_0}}\left(1 + \sqrt{\frac{P_I}{Q_k P_c}}\right)\right] + Q\left[\sqrt{\frac{2E_b}{N_0}}\left(1 - \sqrt{\frac{P_I}{Q_k P_c}}\right)\right]\right\}. \qquad (3.28)$$

To make this applicable to our system, we will use the identity of the probability of symbol error for QPSK where

$$P_{S,QPSK} = 2P_{b,BPSK}\left[1 - \frac{1}{2}P_{b,BPSK}\right]. \qquad (3.29)$$

This new equation, when compared with the simulation plots, yielded some very interesting similarities. Figure 3.5 illustrates the comparison between this equation for QPSK and the simulation. For all cases E_b/N_0 was set so that the probability of symbol error with only AWGN was 10^{-5}. As the figure shows, (3.28) produced results that were very good for spreading factors of $Q_k = 4,16$. For values of $Q_k = 2,8$ the equation did not hold, but looked more like the theoretical results obtained with (3.25).

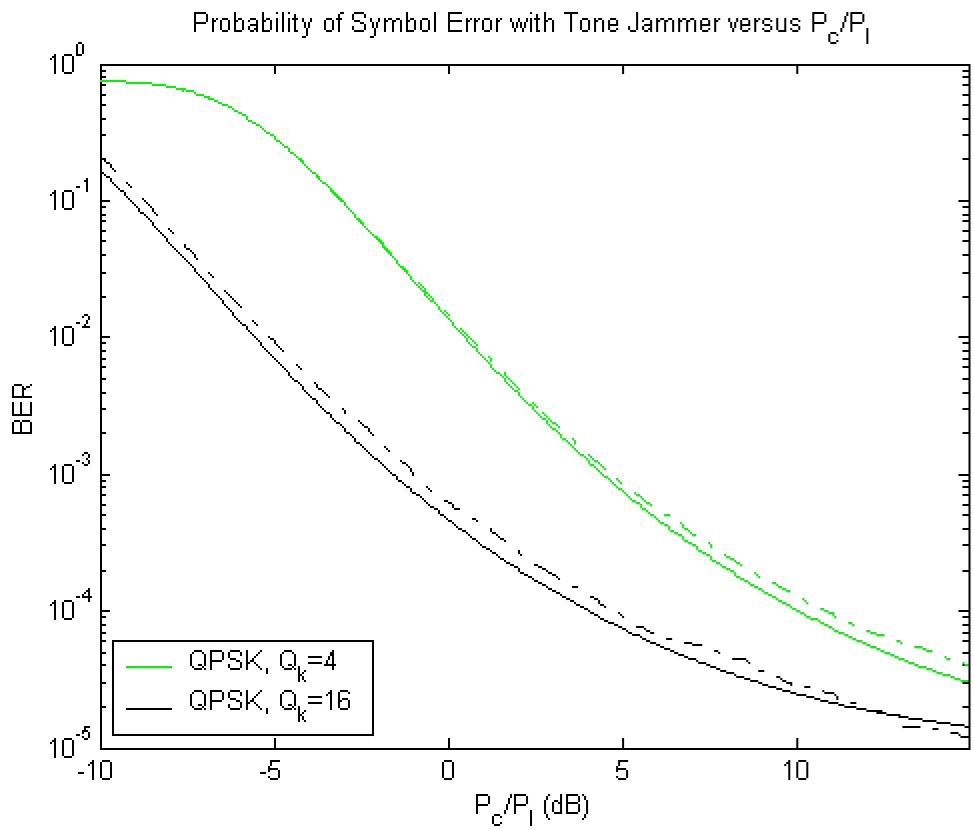

Figure 3.5. Probability of symbol error for QPSK and tone jamming. The solid lines are the theoretical solutions and the dot-dashed lines are the simulation results.

2. Barrage Jamming

Barrage jamming entails transmitting a band-limited signal whose PSD is spectrally flat over the signal bandwidth. This is the same as referring to the signal as band-limited white Gaussian noise. The main drawback of barrage jamming is that this type of jamming requires more average power than tone jamming.

To barrage jam a TD-SCDMA signal, the jamming signal must have a bandwidth equal to 1.6 MHz and be centered at the carrier frequency of the TD-SCDMA signal to be jammed. This allows the barrage noise to completely saturate the spectrum of the transmitted spread signal. If we assume that the barrage-jamming signal is an ideal waveform, then the PSD of the signal is spectrally flat over the passband region and is zero elsewhere. In this analysis we assume that the noise has a PSD of $N_I/2$ as this makes the computations much easier.

As the interference signal enters the TD-SCDMA receiver the heterodyning process creates replicas of the waveform at baseband and twice the carrier frequency. Assuming a two-sided PSD, this means that the signal at baseband has twice the amplitude of the original signal, or a PSD of N_I between $\pm R_c/2$. As we did earlier for AWGN, we can effectively ignore the terms at twice the carrier frequency since the root-raised cosine filter is a low-pass filter.

At this point we see that the system creates a noise term whose spectral characteristics are the same as if we had AWGN. This allows us to treat the barrage-jamming signal like we did with thermal noise and yields

$$P_M \approx 2Q\left(\sqrt{\frac{2 \cdot \log_2(M) E_b}{N_0 + N_I}} \sin\left(\frac{\pi}{M}\right)\right) \tag{3.30}$$

where N_I is the additive noise PSD due to the interfering barrage jamming signal. Again this equation is not exact for small values of M, and a more accurate equation for the probability of symbol error and QPSK can be found using

$$P_4 = 2Q\left(\sqrt{\frac{2E_b}{N_0 + N_I}}\right)\left[1 - \frac{1}{2}Q\left(\sqrt{\frac{2E_b}{N_0 + N_I}}\right)\right]. \tag{3.31}$$

When we simulate the effect of barrage noise jamming, we obtain results as shown in Figure 3.6. For all cases E_b/N_0 was set so that the probability of symbol error with only AWGN was 10^{-5}. As can be seen in the Figure 3.6, the simulation results match well with the theoretical results and approach the noise floor of 10^{-5} as expected. Again, the slight error at larger signal-to-interference ratios is due in part to the author not exactly able to set the ratio of E_b/N_I against the root-raised cosine filter.

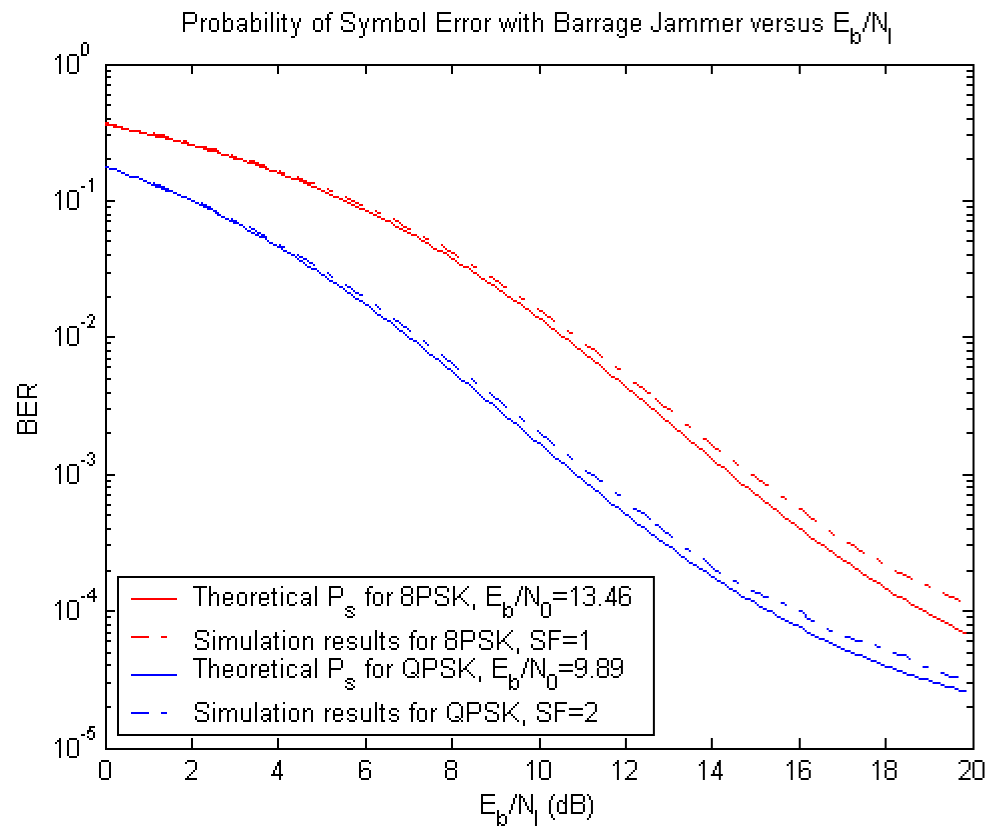

Figure 3.6. Probability of symbol error for QPSK and 8PSK with barrage noise jamming.

Although it is not shown, spreading had minimal effect on the above simulation. We can show that for QPSK with spreading factors of 4, 8, and 16 the simulation produces results similar to those shown for a spreading factor of 2. Because they are almost indistinguishable they were not shown in order to make the plot more discernable.

IV. INTERCEPTION AND EXPLOITATION

One of the more interesting aspects of this thesis was to analyze a way to covertly intercept a TD-SCDMA transmission. Although the standard provides a lawful interception procedure, it is strictly controlled and only authorized personnel are allowed to utilize this service. This implies that the service provider has knowledge of who is being monitored and what government agency is requesting the interception. To be truly covert a method to intercept the desired signal without a middleman is needed.

1. Lawful Interception

The national and international laws of the involved countries tightly control lawful interception of TD-SCDMA transmissions. Only certain agencies under certain conditions can request the interception of a target's transmission and only for a certain period of time.

Lawful interception is a very straightforward procedure and involves decryption of the intended target's transmissions at the base station. This procedure intercepts the transmission after descrambling, despreading, and demodulation. This is analogous to a mobile user calling a person on a landline telephone. The cellular signal must be demodulated down to baseband at the base station and converted to a signal compatible with a household telephone. At this point, if the transmission is a voice signal we can see that a mere tape recorder can be tapped into the line to capture the signal. For packet data transmissions, a more elaborate technique may be necessary.

The hardest part of deciphering the signal may be if the user encrypted the information data before transmission. In this case, the agency requesting the target's transmissions is responsible for breaking the code.

Although lawful interception is available, certain conditions may arise where time is of the essence and obtaining permission may be lengthy and burdensome. In those cases covert interception could be employed.

2. Covert Interception

As mentioned before, one of TD-SCDMA's most powerful assets is that the system uses TDD to utilize the same physical channel for both uplink and downlink transmissions. Although commercially this is advantageous, this makes TD-SCDMA much easier to exploit. Given that the center carrier frequency for each physical channel is a multiple of 200 kHz and the frequency bands are published, this makes identification and detection of a two-way TD-SCDMA transmission much easier.

Even though a signal can be detected, the key to intercepting and making sense of the transmission depends on synchronization. If the interceptor is not time-aligned to the detected signal, then everything will be deciphered as garbage. Luckily, the standard for TD-SCDMA details the procedure for user synchronization.

a. User Synchronization

When the user activates his or her handset, the handset begins a cell search procedure to locate a nearby base station. The search begins by locating the downlink pilot tone signal (DwPTS) on the SYNC-DL channel from any primary common control physical channel (P-CCPCH). This is accomplished by the use of matched filters and is made easier by the fact that the SYNC-DL and SYNC-UL channels are not spread or scrambled. Since the pilot tone signal is not coded and always repeats itself every couple of radio frames, the user runs a correlation until the DwPTS is detected. The only difficulty lies in the fact that there are 32 different DwPTS codes and the handset must try them all until the correct one is found.

After the handset has located and synchronized to the SYNC-DL channel, the next procedure is to identify the cell specific midamble and scrambling codes. Because each SYNC-DL is associated with only four of the available 128 midamble codes, the handset uses a trial and error procedure to find the correct midamble code. Again, the midamble codes are not spread or scrambled and thus are easier to detect. To make this entire procedure even simpler, each scrambling code is associated with a particular midamble code and, therefore, detecting the correct midamble code automatically gives the user the corresponding scrambling code.

After the handset has established the downlink synchronization, the mobile sends an uplink pilot tone signal (UpPTS) on the SYNC-UL channel to begin uplink synchronization. This pilot signal is necessary to allow the base station to adjust the uplink timing of the user handset. Without uplink synchronization, the orthogonality of multiple users on the same physical channel cannot be assured at the base station. This generally leads to increased multi-user interference and an increased bit error rate for all users.

The main cause of signals arriving at the base station out of synchronization is that each mobile is more than likely at a different distance from the base station. The propagation time that is needed to cover these distances causes the transmitted signals from the mobiles to the base station to be slightly delayed or advanced with respect to each other. The delays from the multiple users will cause the received signals at the base station to be out of synchronization with each other, and the Walsh functions are not orthogonal if they are not time aligned. This is much less of a problem between the users themselves because the uplink signals are at a much lower transmitter power level. On the downlink, asynchronous transmissions are also much less of a problem since the base station transmits all user signals at the same time, and in the absence of multi-path fading they arrive synchronous to each other at a given user.

Once the SYNC-UL signal is received, the base station computes the propagation delay, and a synchronization shift (SS) symbol is sent on the next available downlink timeslot. This SS symbol instructs the user handset to advance or retard the uplink transmissions to ensure that all user transmissions arrive at the base station synchronous to each other. The base station and user repeat the synchronization procedure every radio frame on the P-CCPCH.

Now that the handset is synchronized and has the scrambling and midamble codes, all that remains is to read the master information block (MIB), register with the base station network, monitor the broadcast channel (BCH), and locate the control channels. Since the P-CCPCH contains all this information, the procedure is relatively straightforward. In particular, the handset will monitor the paging channel to

know when an incoming call is being transmitted and periodically check the DwPTS to maintain synchronization.

b. Interception

To intercept a TD-SCDMA transmission, the covert interceptors must first synchronize their device to the DwPTS from the base station and identify the appropriate cell-specific midamble and scrambling codes as described above. This aspect is essential since the intended target's handset is also time aligned to the same transmission. Even though there may be different propagation delays between the base station and the two handsets (interceptor and target), this synchronization ensures that any received signal from the base station is time aligned. Because the covert handset does not send any transmissions to the base station, the interceptor need not worry about establishing any uplink synchronization with the base station.

Once the downlink synchronization is accomplished, the interceptor must search for and synchronize to the UpPTS signal from the desired target. This is the hardest task and requires that the interceptor know or be able to obtain the target's identification number. Without this information intercepting the target will be almost impossible.

Synchronization with the target is made easier by the fact that once the interceptor has obtained the SYNC-DL code, there are only eight unique SYNC-UL codes associated with that SYNC-DL code. The only problem again lies in the fact that all users in the same cell are using the same eight SYNC-UL codes. Unless the target's identification is known, there is no way to know which channel to monitor and at what point in the radio frame to check.

Of course, this is a simplified procedure to obtain synchronization between the interceptor, base station, and target. Once this is accomplished, the task of interpreting all the control signals and acting on them is beyond the scope of this thesis. The main scope of this study was the physical layer of TD-SCDMA and upper layer procedures were not researched in-depth. In theory though, interception and exploitation can be accomplished once synchronization is established but will require a more in-depth study into the radio resource control architecture.

V. CONCLUSIONS AND RECOMMENDATIONS

A. CONCLUSIONS

Currently, TD-SCDMA has been incorporated into the 3G cellular standard of UMTS/UTRA as the TDD low chiprate option. This 3G standard, which includes WCDMA, is based on the GSM core network that is prevalent throughout much of the world. TD-SCDMA had previously been submitted as a separate candidate, but since the goal of IMT-2000 was to harmonize the world with a global cellular standard, several candidates, including TD-SCDMA, were incorporated into others. Nevertheless, there is no stipulation that TD-SCDMA cannot be employed as a separate system since the ITU has no legal authority to enforce their ideals of global standardization. For example, China, where TD-SCDMA was originally developed, is a prime candidate for supporting and utilizing their homegrown technology. Even so, it is more than likely that service providers of 2G and 2.5G cellular service will chose an upgrade path to 3G which allows backward compatibility with their current systems.

The main advantage of using TD-SCDMA is that the system employs TDD to have both the uplink and downlink capabilities on the same frequency band. This is a departure from WCDMA and CDMA2000, which employ FDD and a paired spectrum for forward and reverse links. Using an unpaired spectrum allows TD-SCDMA to be more versatile than those systems that do not employ this technology. As a comparison, three TD-SCDMA physical channels can fit within one WCDMA paired spectrum or two CDMA2000 paired spectrums. Although this can be advantageous at times, TD-SCDMA generally has a lower user/cell ratio than the other two given the same bandwidth.

The second main feature of TD-SCDMA is that this technology utilizes code division multiple access, as the name implies. This means that in addition to using one physical channel for both uplink and downlink transmissions, multiple users can share the same physical channel with only minimal interference between their signals. Therefore, like IS-95, TD-SCDMA is an interference-limited system where the more users on a given physical channel the greater the multi-user interference.

When analyzed, the performance of TD-SCDMA under adverse conditions is very similar to other CDMA systems. A slight difference lies in the fact that TD-SCDMA employs matching root-raised cosine filters at both the transmitter and receiver to reduce ISI. The exact details of how this filter performed were not well documented and the author had to derive them from start to finish, which was made more difficult due to the fact that TD-SCDMA employs complex valued spreading codes. Interestingly, having this type of filter at the receiver, even though the filter is low-pass in nature, has minimal effect on AWGN aside from a scaling factor. This phenomenon was computed analytically and verified via simulation. Implementation of this property made further analysis much easier and allowed the author to use existing analytical equations to verify the simulations.

Because AWGN is not an interesting or challenging adverse condition, Rayleigh fading was also considered and additional simulation results were produced. Again, the results were similar to existing theoretical equations since TD-SCDMA is akin to most direct sequence spread spectrum systems. Although the plots obtained when employing the RRC seem to show a relationship between the spreading factor used and the BER, this could not be shown analytically. Rerunning the simulation without the RRC produced the expected results and implies that the author has an error in establishing the proper average signal-to-noise ratio.

The next step was to examine attempts to disrupt the simulated TD-SCDMA transmission. Two types of interfering methods were employed, tone jamming and barrage-jamming. Because the performance of TD-SCDMA is similar to other CDMA systems, the results of jamming the transmission were very close. Only the tone-jamming scenario produced significantly different results, but this was due to the fact that TD-SCDMA employs very small spreading factors and the only theoretical equations that exist are for large spreading factors. We found that for small spreading factors the tone-jamming signal has an affect similar to a standard non-CDMA QPSK signal with tone jamming. By adding a constant related to the spreading factor, the author was able to match an analytical equation to the numerical simulations. This could not be verified for all spreading factors, but enough compared relatively well that these theoretical equations have some merit.

The last stage was to theorize how an individual or organization could intercept and exploit a TD-SCDMA transmission. Because all 3G standards are published and well documented, the radio protocol processes can be adapted to allow for covert interception of TD-SCDMA transmissions. The key to the process is synchronization, and without this fundamental aspect no interception can take place. Once this synchronization is achieved, with both the base station and the intended target, then more detailed procedures can be developed to intercept and interpret the signals.

All things considered, despite TD-SCDMA's advantage by using TDD and having a smaller bandwidth, the author does not foresee this technology gaining a large share of the 3G market unless China goes with their homegrown system. WCDMA and CDMA2000, the two main competitors, already have a solid core network and marketing base in various parts of the world. This means that TD-SCDMA, which uses principles employed by both WCDMA and CDMA2000, will have a hard time attracting customers. This can be seen in the fact that TD-SCDMA was incorporated in UTRA as a low chiprate *option* whereas WCDMA is more prominent in that standard. Without a key feature that will improve the performance of a system above and beyond WCDMA or CDMA2000, the author doubts any other system will gain much popularity.

B. RECOMMENDATIONS

For further analysis, the author recommends that FEC coding be evaluated and incorporated into the simulation. This will produce more useful results instead of the probability of symbol error plots that are presented within this thesis. Although the probability of symbol error is needed to compute the probability of bit error, the end user is more concerned with the number of errors that will occur at the final output of the receiver.

Another recommendation is that the simulation be re-evaluated when computing the probability of symbol error in the presence of Rayleigh fading. Although the results are comparable with theoretical solutions when performing the simulation without RRC filtering, the results *with* RRC filtering should be so as well.

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX A. MATLAB CODE FOR TD-SCDMA IN AWGN.

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% TD-SCDMA in additive white Gaussian noise (AWGN)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
clear
Qk=1;% Spreading Factor (SF)
if Qk==1;
   modu=1;M=8;% use 8PSK
else;
   modu=0;M=4;% use QPSK
end;
Qmax=16;% maximum spreading factor
k=1;% user number in a time slot
Kmax=16;% maximum number of users in a time slot
chiprate=1.28*10^6;% chiprate
Tc=1/chiprate;% chip duration = 1 / chiprate
cell=0;% cell number for scrambling code (could be random)
if Qk==1;
   c=[1]; % Orthogonal Variable Spreading Factor (OVSF)
   Nk=352; % Number of data points
   w=[1]; % weight multiplier
elseif Qk==2;
   c=[1 1;1 -1];Nk=176;w=[1 i];
elseif Qk==4;
   c=[1 1 1 1;1 1 -1 -1;1 -1 1 -1;1 -1 -1 1];Nk=88;w=[-i 1 i -1];
elseif Qk==8;
   c=[ 1 1 1 1 1 1 1 1;1 1 1 1 -1 -1 -1 -1;1 1 -1 -1 1 1 -1 -1;1 1 -1 -1 -1 -1 1 1;
      1 -1 1 -1 1 -1 1 -1;1 -1 1 -1 -1 1 -1 1;1 -1 -1 1 1 -1 -1 1;1 -1 -1 1 -1 1 1 -1];
   Nk=44;w=[1 i i -1 -i -1 -i 1];
else Qk==16;
   c=[ 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1;1 1 1 1 1 1 1 1 -1 -1 -1 -1 -1 -1 -1 -1;
      1 1 1 1 -1 -1 -1 -1 1 1 1 1 -1 -1 -1 -1;1 1 1 1 -1 -1 -1 -1 -1 -1 -1 -1 1 1 1 1;
      1 1 -1 -1 1 1 -1 -1 1 1 -1 -1 1 1 -1 -1;1 1 -1 -1 1 1 -1 -1 -1 -1 1 1 -1 -1 1 1;
      1 1 -1 -1 -1 -1 1 1 1 1 -1 -1 -1 -1 1 1;1 1 -1 -1 -1 -1 1 1 -1 -1 1 1 1 1 -1 -1;
      1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1;1 -1 1 -1 1 -1 1 -1 -1 1 -1 1 -1 1 -1 1;
      1 -1 1 -1 -1 1 -1 1 1 -1 1 -1 -1 1 -1 1;1 -1 1 -1 -1 1 -1 1 -1 1 -1 1 1 -1 1 -1;
      1 -1 -1 1 1 -1 -1 1 1 -1 -1 1 1 -1 -1 1;1 -1 -1 1 1 -1 -1 1 -1 1 1 -1 -1 1 1 -1;
      1 -1 -1 1 -1 1 1 -1 1 -1 -1 1 -1 1 1 -1;1 -1 -1 1 -1 1 1 -1 -1 1 1 -1 1 -1 -1 1];
   Nk=22;w=[-1 -i 1 1 i -1 -1 1 -i i 1 i -i -i i -1];
end
% Generate data sequence
temp=randint(1,Nk,M);
if modu==1;
   do=cos(pi/8+2*temp.*pi/8)+i*sin(pi/8+2*temp.*pi/8);% random 8PSK data sequence
```

```
else;
   do=i.^temp;% random QPSK data sequence
end;
% partial set of scrambling codes
v=[-1 1 -1 -1 -1 1 -1 -1 1 -1 1 1 -1 1 -1 -1;1 1 1 1 1 -1 1 -1 1 -1 -1 1 1 1 -1 -1;
   1 -1 1 1 1 -1 1 1 -1 1 1 1 1 -1 -1 -1;1 1 1 -1 -1 -1 -1 1 -1 -1 1 -1 -1 -1 1 -1;
   1 1 1 -1 -1 -1 -1 1 1 1 1 -1 1 1 1 -1;-1 1 1 -1 -1 -1 1 1 1 1 1 1 1 -1 1 -1;
   -1 1 -1 -1 -1 1 -1 -1 -1 1 1 1 1 -1 -1 -1;1 -1 1 -1 -1 -1 -1 -1 1 1 -1 -1 -1 1 1 -1;
   1 1 1 -1 -1 -1 1 -1 1 1 -1 1 1 1 1 -1;1 1 -1 1 1 1 1 -1 1 1 1 -1 -1 -1 1 -1;
   1 -1 1 -1 1 1 1 1 -1 -1 1 1 -1 1 1 -1;-1 1 1 1 1 -1 -1 -1 -1 1 -1 -1 -1 1 -1 -1;
   -1 -1 1 -1 -1 -1 1 -1 -1 -1 -1 1 1 1 1 -1;1 -1 1 1 1 -1 -1 -1 1 -1 -1 -1 -1 1 -1 -1;
   1 -1 -1 -1 -1 1 -1 -1 1 -1 1 1 1 -1 -1 -1;1 1 -1 -1 -1 1 1 -1 1 -1 1 -1 -1 -1 -1 -1;
   1 -1 -1 1 -1 1 -1 1 -1 -1 -1 -1 1 1 -1 -1;1 1 1 -1 1 1 1 -1 1 1 -1 1 -1 -1 1 -1;
   -1 1 1 1 -1 1 -1 -1 -1 1 -1 -1 1 -1 -1 -1;-1 1 -1 -1 1 -1 -1 -1 -1 1 1 1 -1 1 -1 -1;
   -1 -1 -1 -1 1 -1 1 -1 -1 1 1 -1 1 1 -1 -1;1 1 1 1 -1 -1 1 1 -1 1 1 -1 1 -1 1 -1;
   1 -1 -1 -1 -1 1 1 1 -1 1 -1 -1 -1 1 -1 -1;-1 1 1 1 -1 1 1 1 1 -1 1 1 -1 1 -1 -1;
   -1 -1 1 -1 1 1 1 -1 -1 -1 -1 1 -1 -1 1 -1;1 -1 1 1 1 -1 1 1 1 -1 1 1 -1 1 -1 -1;
   1 -1 -1 -1 1 -1 -1 -1 -1 1 1 1 1 -1 -1 -1;-1 1 -1 -1 -1 1 1 1 1 -1 -1 -1 -1 1 -1 -1;
   -1 -1 -1 1 -1 -1 -1 1 -1 -1 -1 1 1 1 1 -1;1 -1 1 1 -1 1 -1 -1 -1 1 -1 -1 -1 1 -1 -1;
   -1 -1 -1 -1 -1 -1 1 1 1 -1 -1 1 1 -1 1 -1;1 1 -1 -1 1 1 1 1 -1 1 -1 1 -1 1 1 -1;
   1 -1 -1 -1 1 -1 1 1 -1 1 -1 -1 1 -1 -1 -1;-1 -1 -1 1 1 1 1 -1 1 1 1 -1 1 1 1 -1;
   1 -1 -1 -1 1 -1 -1 -1 1 -1 1 1 -1 1 -1 -1;1 -1 1 1 -1 1 -1 -1 1 -1 -1 -1 1 -1 -1 -1;
   1 1 -1 1 1 1 -1 1 -1 -1 -1 1 1 1 1 -1;-1 -1 -1 1 -1 -1 1 -1 -1 -1 1 -1 1 1 1 -1;
   -1 1 -1 -1 1 -1 1 1 1 -1 -1 -1 1 -1 -1 -1;-1 1 1 1 1 -1 -1 -1 1 -1 -1 -1 1 -1 -1 -1;
   -1 1 -1 1 -1 -1 -1 -1 -1 -1 1 1 -1 1 1 -1;1 1 -1 1 -1 -1 1 -1 -1 -1 1 -1 -1 -1 1 -1;
   1 -1 -1 -1 -1 1 1 1 1 -1 -1 -1 1 -1 -1 -1;-1 -1 1 1 -1 -1 -1 -1 -1 1 -1 1 -1 1 1 -1;
   -1 -1 1 -1 -1 -1 -1 1 1 1 1 -1 -1 -1 1 -1;-1 -1 1 -1 1 1 -1 1 1 1 1 -1 1 1 1 -1];
% Perform spreading and scrambling of data sequence
block=[];
for n=1:Nk;
   for q=1:Qk;
      s=c(k,1+mod((n-1)*Qk+q-1,Qk))*(i^(1+mod((n-1)*Qk+q-1,Qmax))*...
         v(cell+1,1+mod((n-1)*Qk+q-1,Qmax)));
      block(1,Qk*(n-1)+q)=do(1,n)*w(1,k)*s;
   end;
end;
% filter chipped sequence with root-raised cosine (RRC)
Delay=3;DataL=Nk*Qk;R=0.22;S=8;Fs=S*chiprate;Fd=chiprate;PropD=Delay/Fd;
x=block';
tx=[PropD:PropD+DataL-1]./Fd;
xreal=real(x);ximag=imag(x);
[yrealt,trealt]=rcosflt(xreal,Fd,Fs,'sqrt',R,Delay);% matching RRC filter
[yimagt,timagt]=rcosflt(ximag,Fd,Fs,'sqrt',R,Delay);% matching RRC filter
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Baseband Receiver
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
w1=conj(w);% to compensate for complex multiplier
probt=0;% ensure a clear probability of bit error counter
num=1;% number of iterations to run simulation
if Qk==1;pt=135;elseif Qk==2;pt=105;elseif Qk==4;pt=105;...
   elseif Qk==8;pt=105;else;pt=105;end;
for u=1:num;
   noise1=randn(length(yrealt),1)*sqrt(Qk/(2*log2(M)));% generate in-phase noise
   noise2=randn(length(yimagt),1)*sqrt(Qk/(2*log2(M)));% generate quadrature-phase
noise
   for r=1:pt;
      yrealc=yrealt+sqrt(1/(10^((r-1)/100)))*noise1;% add noise to signal
      yimagc=yimagt-sqrt(1/(10^((r-1)/100)))*noise2;% add noise to signal
      trealc=trealt;timagc=timagt;
      [yrealr,trealr]=rcosflt(yrealc,Fd,Fs,'sqrt/Fs');% matching RRC filter
      [yimagr,timagr]=rcosflt(yimagc,Fd,Fs,'sqrt/Fs');% matching RRC filter
      yrec=yrealr+i*yimagr;yrec=yrec';
      L=length(yrealr)-length(yrealc)+1-S;% offset for time delay
      % sample RRC signal while despreading and descrambling
      block2=[];
      for n=1:Nk;
         for q=1:Qk;
            s=c(k,1+mod((n-1)*Qk+q-1,Qk))*((-i)^(1+mod((n-1)*Qk+q-1,Qmax))*...
               v(cell+1,1+mod((n-1)*Qk+q-1,Qmax)));
            block2(1,Qk*(n-1)+q)=yrec(1,S*(Qk*(n-1)+q)+L)*w1(1,k)*s;
         end;
      end;
      % use summer to recombine spread signal (could use integrator)
      block2=reshape(block2,Qk,Nk);
      if Qk==1;
         block2=block2;
      else;
         block2=sum(block2)/Qk;
      end;
      % compute arctangent to reconstruct MPSK signal
      if modu==1;
         for n=1:Nk;
            ytanr=atan2(imag(block2(1,n)),real(block2(1,n)));
            if ytanr < -3*pi/4;
               yr(1,n)=cos(9*pi/8)+i*sin(9*pi/8);
            elseif ytanr < -pi/2;
               yr(1,n)=cos(11*pi/8)+i*sin(11*pi/8);
            elseif ytanr < -pi/4;
               yr(1,n)=cos(13*pi/8)+i*sin(13*pi/8);
            elseif ytanr < 0;
               yr(1,n)=cos(15*pi/8)+i*sin(15*pi/8);
            elseif ytanr < pi/4;
```

```
            yr(1,n)=cos(pi/8)+i*sin(pi/8);
          elseif ytanr < pi/2;
            yr(1,n)=cos(3*pi/8)+i*sin(3*pi/8);
          elseif ytanr < 3*pi/4;
            yr(1,n)=cos(5*pi/8)+i*sin(5*pi/8);
          else;
            yr(1,n)=cos(7*pi/8)+i*sin(7*pi/8);
          end;
        end;
      else;
        for n=1:Nk;
          ytanr=atan2(imag(block2(1,n)),real(block2(1,n)));
          if ytanr < -3*pi/4;
            yr(1,n)=-1;
          elseif ytanr < -pi/4;
            yr(1,n)=-i;
          elseif ytanr < pi/4;
            yr(1,n)=1;
          elseif ytanr < 3*pi/4;
            yr(1,n)=i;
          else;
            yr(1,n)=-1;
          end;
        end;
      end;
      prob(1,r)=sum(abs(do-yr)>10^-10)/Nk;
    end;
    probt=probt+prob;
end;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% generate plots
r=1:pt;
pe2=1/2*erfc(sqrt(10.^((r-1)/100)));
pe4=2*pe2.*(1-.5.*pe2);
pe8=erfc(sqrt(log2(M)*10.^((r-1)/100))*sin(pi/M));
probj=probt/num;
figure(1);
t=0:.1:(pt-1)/10;
if M==2;semilogy(t,pe2,'r');
elseif M==4;semilogy(t,pe4,'r');
else;semilogy(t,pe8,'r');
end;
hold on;
semilogy(t,probj,'b');hold off;
axis([0 14.5 10^-5 10^0]);
if Qk==1;title(['Probability of Symbol Error versus E_b/N_0, 8PSK and SF=1']);
```

```
elseif Qk==2;title(['Probability of Symbol Error versus E_b/N_0, QPSK and SF=2']);
elseif Qk==4;title(['Probability of Symbol Error versus E_b/N_0, QPSK and SF=4']);
elseif Qk==8;title(['Probability of Symbol Error versus E_b/N_0, QPSK and SF=8']);
else;title(['Probability of Symbol Error versus E_b/N_0, QPSK and SF=16']);
end;
xlabel('E_b/N_0');ylabel('BER');
```

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX B. MATLAB CODE FOR TD-SCDMA WITH RAYLEIGH FADING.

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% TD-SCDMA in additive white Gaussian noise (AWGN) with Rayleigh fading
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
clear
Qk=1;% Spreading Factor (SF)
if Qk==1;
   modu=1;M=8;% use 8PSK
else;
   modu=0;M=4;% use QPSK
end;
Qmax=16;% maximum spreading factor
k=1;% user number in a time slot
Kmax=16;% maximum number of users in a time slot
chiprate=1.28*10^6;% chiprate
Tc=1/chiprate;% chip duration = 1 / chiprate
cell=0;% cell number for scrambling code (could be random)
if Qk==1;
   c=[1]; % Orthogonal Variable Spreading Factor (OVSF)
   Nk=352; % Number of data points
   w=[1]; % weight multiplier
elseif Qk==2;
   c=[1 1;1 -1];Nk=176;w=[1 i];
elseif Qk==4;
   c=[1 1 1 1;1 1 -1 -1;1 -1 1 -1;1 -1 -1 1];Nk=88;w=[-i 1 i -1];
elseif Qk==8;
   c=[ 1 1 1 1 1 1 1 1;1 1 1 1 -1 -1 -1 -1;1 1 -1 -1 1 1 -1 -1;1 1 -1 -1 -1 -1 1 1;
      1 -1 1 -1 1 -1 1 -1;1 -1 1 -1 -1 1 -1 1;1 -1 -1 1 1 -1 -1 1;1 -1 -1 1 -1 1 1 -1];
   Nk=44;w=[1 i i -1 -i -1 -i 1];
else Qk==16;
   c=[ 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1;1 1 1 1 1 1 1 1 -1 -1 -1 -1 -1 -1 -1 -1;
      1 1 1 1 -1 -1 -1 -1 1 1 1 1 -1 -1 -1 -1;1 1 1 1 -1 -1 -1 -1 -1 -1 -1 -1 1 1 1 1;
      1 1 -1 -1 1 1 -1 -1 1 1 -1 -1 1 1 -1 -1;1 1 -1 -1 1 1 -1 -1 -1 -1 1 1 -1 -1 1 1;
      1 1 -1 -1 -1 -1 1 1 1 1 -1 -1 -1 -1 1 1;1 1 -1 -1 -1 -1 1 1 -1 -1 1 1 1 1 -1 -1;
      1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1;1 -1 1 -1 1 -1 1 -1 -1 1 -1 1 -1 1 -1 1;
      1 -1 1 -1 -1 1 -1 1 1 -1 1 -1 -1 1 -1 1;1 -1 1 -1 -1 1 -1 1 -1 1 -1 1 1 -1 1 -1;
      1 -1 -1 1 1 -1 -1 1 1 -1 -1 1 1 -1 -1 1;1 -1 -1 1 1 -1 -1 1 -1 1 1 -1 -1 1 1 -1;
      1 -1 -1 1 -1 1 1 -1 1 -1 -1 1 -1 1 1 -1;1 -1 -1 1 -1 1 1 -1 -1 1 1 -1 1 -1 -1 1];
   Nk=22;w=[-1 -i 1 1 i -1 -1 1 -i i 1 i -i -i i -1];
end
% Generate data sequence
temp=randint(1,Nk,M);
if modu==1;
```

```
   do=cos(pi/8+2*temp.*pi/8)+i*sin(pi/8+2*temp.*pi/8);% random 8PSK data sequence
else;
   do=i.^temp;% random QPSK data sequence
end;
% partial set of scrambling codes
v=[-1 1 -1 -1 -1 1 -1 -1 1 -1 1 1 -1 1 -1 -1;1 1 1 1 1 -1 1 -1 1 -1 -1 1 1 1 -1 -1;
   1 -1 1 1 1 -1 1 1 -1 1 1 1 1 -1 -1 -1;1 1 1 -1 -1 -1 -1 1 -1 -1 1 -1 -1 -1 1 -1;
   1 1 1 -1 -1 -1 -1 1 1 1 1 -1 1 1 1 -1;-1 1 1 -1 -1 -1 1 1 1 1 1 1 1 -1 1 -1;
   -1 1 -1 -1 -1 1 -1 -1 -1 1 1 1 1 -1 -1 -1;1 -1 1 -1 -1 -1 -1 -1 1 1 -1 -1 -1 1 1 -1;
   1 1 1 -1 -1 -1 1 -1 1 1 -1 1 1 1 1 -1;1 1 -1 1 1 1 1 -1 1 1 1 -1 -1 -1 1 -1;
   1 -1 1 -1 1 1 1 1 -1 -1 1 1 -1 1 1 -1;-1 1 1 1 1 -1 -1 -1 -1 1 -1 -1 -1 1 -1 -1;
   -1 -1 1 -1 -1 -1 1 -1 -1 -1 -1 1 1 1 1 -1;1 -1 1 1 1 -1 -1 -1 1 -1 -1 -1 -1 1 -1 -1;
   1 -1 -1 -1 -1 1 -1 -1 1 -1 1 1 1 -1 -1 -1;1 1 -1 -1 -1 1 1 -1 1 -1 1 -1 -1 -1 -1 -1;
   1 -1 -1 1 -1 1 -1 1 -1 -1 -1 -1 1 1 -1 -1;1 1 1 -1 1 1 1 -1 1 1 -1 1 -1 -1 1 -1;
   -1 1 1 1 -1 1 -1 -1 -1 1 -1 -1 1 -1 -1 -1;-1 1 -1 -1 1 -1 -1 -1 -1 1 1 1 -1 1 -1 -1;
   -1 -1 -1 -1 1 -1 1 -1 -1 1 1 -1 1 1 -1 -1;1 1 1 1 -1 -1 1 1 -1 1 1 -1 1 -1 1 -1;
   1 -1 -1 -1 -1 1 1 1 -1 1 -1 -1 -1 1 -1 -1;-1 1 1 1 -1 1 1 1 1 -1 1 1 -1 1 -1 -1;
   -1 -1 1 -1 1 1 1 -1 -1 -1 -1 1 -1 -1 1 -1;1 -1 1 1 1 -1 1 1 1 -1 1 1 -1 1 -1 -1;
   1 -1 -1 -1 1 -1 -1 -1 -1 1 1 1 1 -1 -1 -1;-1 1 -1 -1 -1 1 1 1 1 -1 -1 -1 -1 1 -1 -1;
   -1 -1 -1 1 -1 -1 -1 1 -1 -1 -1 1 1 1 1 -1;1 -1 1 1 -1 1 -1 -1 -1 1 -1 -1 -1 1 -1 -1;
   -1 -1 -1 -1 -1 -1 1 1 1 -1 -1 1 1 -1 1 -1;1 1 -1 -1 1 1 1 1 -1 1 -1 1 -1 1 1 -1;
   1 -1 -1 -1 1 -1 1 1 -1 1 -1 -1 1 -1 -1 -1;-1 -1 -1 1 1 1 1 -1 1 1 1 -1 1 1 1 -1;
   1 -1 -1 -1 1 -1 -1 -1 1 -1 1 1 -1 1 -1 -1;1 -1 1 1 -1 1 -1 -1 1 -1 -1 -1 1 -1 -1 -1;
   1 1 -1 1 1 1 -1 1 -1 -1 -1 1 1 1 1 -1;-1 -1 -1 1 -1 -1 1 -1 -1 -1 1 -1 1 1 1 1 -1;
   -1 1 -1 -1 1 -1 1 1 1 -1 -1 -1 1 -1 -1 -1;-1 1 1 1 1 -1 -1 -1 1 -1 -1 -1 1 -1 -1 -1;
   -1 1 -1 1 -1 -1 -1 -1 -1 -1 1 1 -1 1 1 -1;1 1 -1 1 -1 -1 1 -1 -1 -1 1 -1 -1 -1 1 -1;
   1 -1 -1 -1 -1 1 1 1 1 -1 -1 -1 1 -1 -1 -1;-1 -1 1 1 -1 -1 -1 -1 -1 1 -1 1 -1 1 1 -1;
   -1 -1 1 -1 -1 -1 -1 1 1 1 1 -1 -1 -1 1 -1;-1 -1 1 -1 1 1 -1 1 1 1 1 -1 1 1 1 -1];
% Perform spreading and scrambling of data sequence
block=[];
for n=1:Nk;
   for q=1:Qk;
      s=c(k,1+mod((n-1)*Qk+q-1,Qk))*(i^(1+mod((n-1)*Qk+q-1,Qmax))*...
         v(cell+1,1+mod((n-1)*Qk+q-1,Qmax)));
      block(1,Qk*(n-1)+q)=do(1,n)*w(1,k)*s;
   end;
end;
% filter chipped sequence with root-raised cosine (RRC)
Delay=3;DataL=Nk*Qk;R=0.22;S=8;Fs=S*chiprate;Fd=chiprate;PropD=Delay/Fd;
x=block';
tx=[PropD:PropD+DataL-1]./Fd;
xreal=real(x);ximag=imag(x);
[yrealt,trealt]=rcosflt(xreal,Fd,Fs,'sqrt',R,Delay);% matching RRC filter
[yimagt,timagt]=rcosflt(ximag,Fd,Fs,'sqrt',R,Delay);% matching RRC filter

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
%Baseband Receiver
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
w1=conj(w);% to compensate for complex multiplier
probt=0;% ensure a clear probability of bit error counter
num=1;% number of iterations to run simulation
if Qk==1;pt=135;elseif Qk==2;pt=105;elseif Qk==4;pt=105;...
   elseif Qk==8;pt=105;else;pt=105;end;
for u=1:num;
   noise1=randn(length(yrealt),1)*sqrt(Qk/(2*log2(M)));% generate in-phase noise
   noise2=randn(length(yimagt),1)*sqrt(Qk/(2*log2(M)));% generate quadrature-phase
noise
   if Qk==1;h=716;elseif Qk==2;h=358;elseif Qk==4;h=180;...
      elseif Qk==8;h=90;else; h=46;end;
   g=randn(1,352*8+3*8*2)*sqrt(1/2);% generate Gaussian random noise
   pt=140;
   % generate Rayleigh distributed amplitude
   for j=1:2:h+96;
      ray(1,8*Qk*(j-1)/2+1:8*Qk*(j-1)/2+8*Qk)=sqrt(g(j)^2+g(j+1)^2);
   end;
   ray=ray(1,1:length(yrealt));
   for r=1:pt;
      yrealc=yrealt.*ray'+sqrt(1/10^((r-1)/100))*noise1;
      yimagc=yimagt.*ray'+sqrt(1/10^((r-1)/100))*noise2;
      trealc=trealt;timagc=timagt;
      [yrealr,trealr]=rcosflt(yrealc,Fd,Fs,'sqrt/Fs');% matching RRC filter
      [yimagr,timagr]=rcosflt(yimagc,Fd,Fs,'sqrt/Fs');% matching RRC filter
      yrec=yrealr+i*yimagr;yrec=yrec';
      L=length(yrealr)-length(yrealc)+1-S;% offset for time delay
      % sample RRC signal while despreading and descrambling
      block2=[];
      for n=1:Nk;
         for q=1:Qk;
            s=c(k,1+mod((n-1)*Qk+q-1,Qk))*((-i)^(1+mod((n-1)*Qk+q-1,Qmax))*...
               v(cell+1,1+mod((n-1)*Qk+q-1,Qmax)));
            block2(1,Qk*(n-1)+q)=yrec(1,S*(Qk*(n-1)+q)+L)*w1(1,k)*s;
         end;
      end;
      % use summer to recombine spread signal (could use integrator)
      block2=reshape(block2,Qk,Nk);
      if Qk==1;
         block2=block2;
      else;
         block2=sum(block2)/Qk;
      end;
      % compute arctangent to reconstruct MPSK signal
      if modu==1;
```

```
        for n=1:Nk;
            ytanr=atan2(imag(block2(1,n)),real(block2(1,n)));
            if ytanr < -3*pi/4;
                yr(1,n)=cos(9*pi/8)+i*sin(9*pi/8);
            elseif ytanr < -pi/2;
                yr(1,n)=cos(11*pi/8)+i*sin(11*pi/8);
            elseif ytanr < -pi/4;
                yr(1,n)=cos(13*pi/8)+i*sin(13*pi/8);
            elseif ytanr < 0;
                yr(1,n)=cos(15*pi/8)+i*sin(15*pi/8);
            elseif ytanr < pi/4;
                yr(1,n)=cos(pi/8)+i*sin(pi/8);
            elseif ytanr < pi/2;
                yr(1,n)=cos(3*pi/8)+i*sin(3*pi/8);
            elseif ytanr < 3*pi/4;
                yr(1,n)=cos(5*pi/8)+i*sin(5*pi/8);
            else;
                yr(1,n)=cos(7*pi/8)+i*sin(7*pi/8);
            end;
        end;
    else;
        for n=1:Nk;
            ytanr=atan2(imag(block2(1,n)),real(block2(1,n)));
            if ytanr < -3*pi/4;
                yr(1,n)=-1;
            elseif ytanr < -pi/4;
                yr(1,n)=-i;
            elseif ytanr < pi/4;
                yr(1,n)=1;
            elseif ytanr < 3*pi/4;
                yr(1,n)=i;
            else;
                yr(1,n)=-1;
            end;
        end;
    end;
    prob(1,r)=sum(abs(do-yr)>10^-10)/Nk;
  end;
  probt=probt+prob;
end;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% generate plots
r=1:pt;
pe2=1/2*erfc(sqrt(10.^((r-1)/100)));
pe4=erfc(sqrt(10.^((r-1)/100))).*(1-.25*erfc(sqrt(10.^((r-1)/100))));
pe8=erfc(sqrt(log2(M)*10.^((r-1)/100))*sin(pi/M));
```

```
snr=0:.1:(pt-1)/10;z=10.^(snr/10);
Pe=0.5*(1-sqrt(z./(1+z)));
if M==2;Ps=1-sqrt(z./(1+z));
else;Ps=1-sqrt(log2(M)*z*(sin(pi/M))^2./(1+log2(M)*z*(sin(pi/M))^2));
end;
probj=probt/num;
figure(1);
t=0:.1:(pt-1)/10;
if M==2;
   semilogy(t,pe2,'r');
elseif M==4;
   semilogy(t,pe4,'r');
else;
   semilogy(t,pe8,'r');
end;
hold on;
semilogy(t,Ps,'-.g');
semilogy(t,probj,'k');
hold off;
axis([0 14 10^-5 10^0]);
if Qk==1;title(['Pb for Es/No versus Eb/No, 8PSK and SF=1']);
elseif Qk==2;title(['Pb for Es/No versus Eb/No, QPSK and SF=2']);
elseif Qk==4;title(['Pb for Es/No versus Eb/No, QPSK and SF=4']);
elseif Qk==8;title(['Pb for Es/No versus Eb/No, QPSK and SF=8']);
else;title(['Pb for Es/No versus Eb/No, QPSK and SF=16']);
end;
xlabel('E_b/N_0');ylabel('BER');
```

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX C. MATLAB CODE FOR TD-SCDMA WITH TONE JAMMING.

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% TD-SCDMA in additive white Gaussian noise (AWGN) with tone jamming
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
clear
Qk=1;% Spreading Factor (SF)
if Qk==1;
   modu=1;M=8;% use 8PSK
else;
   modu=0;M=4;% use QPSK
end;
Qmax=16;% maximum spreading factor
k=1;% user number in a time slot
Kmax=16;% maximum number of users in a time slot
chiprate=1.28*10^6;% chiprate
Tc=1/chiprate;% chip duration = 1 / chiprate
cell=0;% cell number for scrambling code (could be random)
if Qk==1;
   c=[1]; % Orthogonal Variable Spreading Factor (OVSF)
   Nk=352; % Number of data points
   w=[1]; % weight multiplier
elseif Qk==2;
   c=[1 1;1 -1];Nk=176;w=[1 i];
elseif Qk==4;
   c=[1 1 1 1;1 1 -1 -1;1 -1 1 -1;1 -1 -1 1];Nk=88;w=[-i 1 i -1];
elseif Qk==8;
   c=[ 1 1 1 1 1 1 1 1;1 1 1 1 -1 -1 -1 -1;1 1 -1 -1 1 1 -1 -1;1 1 -1 -1 -1 -1 1 1;
      1 -1 1 -1 1 -1 1 -1;1 -1 1 -1 -1 1 -1 1;1 -1 -1 1 1 -1 -1 1;1 -1 -1 1 -1 1 1 -1];
   Nk=44;w=[1 i i -1 -i -1 -i 1];
else Qk==16;
   c=[ 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1;1 1 1 1 1 1 1 1 -1 -1 -1 -1 -1 -1 -1 -1;
      1 1 1 1 -1 -1 -1 -1 1 1 1 1 -1 -1 -1 -1;1 1 1 1 -1 -1 -1 -1 -1 -1 -1 -1 1 1 1 1;
      1 1 -1 -1 1 1 -1 -1 1 1 -1 -1 1 1 -1 -1;1 1 -1 -1 1 1 -1 -1 -1 -1 1 1 -1 -1 1 1;
      1 1 -1 -1 -1 -1 1 1 1 1 -1 -1 -1 -1 1 1;1 1 -1 -1 -1 -1 1 1 -1 -1 1 1 1 1 -1 -1;
      1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1;1 -1 1 -1 1 -1 1 -1 -1 1 -1 1 -1 1 -1 1;
      1 -1 1 -1 -1 1 -1 1 1 -1 1 -1 -1 1 -1 1;1 -1 1 -1 -1 1 -1 1 -1 1 -1 1 1 -1 1 -1;
      1 -1 -1 1 1 -1 -1 1 1 -1 -1 1 1 -1 -1 1;1 -1 -1 1 1 -1 -1 1 -1 1 1 -1 -1 1 1 -1;
      1 -1 -1 1 -1 1 1 -1 1 -1 -1 1 -1 1 1 -1;1 -1 -1 1 -1 1 1 -1 -1 1 1 -1 1 -1 -1 1];
   Nk=22;w=[-1 -i 1 1 i -1 -1 1 -i i 1 i -i -i i -1];
end
% Generate data sequence
temp=randint(1,Nk,M);
if modu==1;
```

```
   do=cos(pi/8+2*temp.*pi/8)+i*sin(pi/8+2*temp.*pi/8);% random 8PSK data sequence
else;
   do=i.^temp;% random QPSK data sequence
end;
% partial set of scrambling codes
v=[-1 1 -1 -1 -1 1 -1 -1 1 -1 1 1 -1 1 -1 -1;1 1 1 1 1 -1 1 -1 1 -1 -1 1 1 1 -1 -1;
   1 -1 1 1 1 -1 1 1 -1 1 1 1 1 -1 -1 -1;1 1 1 -1 -1 -1 -1 1 -1 -1 1 -1 -1 -1 1 -1;
   1 1 1 -1 -1 -1 -1 1 1 1 1 -1 1 1 1 -1;-1 1 1 -1 -1 -1 1 1 1 1 1 1 1 -1 1 -1;
   -1 1 -1 -1 -1 1 -1 -1 -1 1 1 1 1 -1 -1 -1;1 -1 1 -1 -1 -1 -1 -1 1 1 -1 -1 -1 1 1 -1;
   1 1 1 -1 -1 -1 1 -1 1 1 -1 1 1 1 1 -1;1 1 -1 1 1 1 1 -1 1 1 1 -1 -1 -1 1 -1;
   1 -1 1 -1 1 1 1 1 -1 -1 1 1 -1 1 1 -1;-1 1 1 1 1 -1 -1 -1 -1 1 -1 -1 -1 1 -1 -1;
   -1 -1 1 -1 -1 -1 1 -1 -1 -1 -1 1 1 1 1 -1;1 -1 1 1 1 -1 -1 -1 1 -1 -1 -1 -1 1 -1 -1;
   1 -1 -1 -1 -1 1 -1 -1 1 -1 1 1 1 -1 -1 -1;1 1 -1 -1 -1 1 1 -1 1 -1 1 -1 -1 -1 -1 -1;
   1 -1 -1 1 -1 1 -1 1 -1 -1 -1 -1 1 1 -1 -1;1 1 1 -1 1 1 1 -1 1 1 -1 1 -1 -1 1 -1;
   -1 1 1 1 -1 1 -1 -1 -1 1 -1 -1 1 -1 -1 -1;-1 1 -1 -1 1 -1 -1 -1 -1 1 1 1 -1 1 -1 -1;
   -1 -1 -1 -1 1 -1 1 -1 -1 1 1 -1 1 1 -1 -1;1 1 1 1 -1 -1 1 1 -1 1 1 -1 1 -1 1 -1;
   1 -1 -1 -1 -1 1 1 1 -1 1 -1 -1 -1 1 -1 -1;-1 1 1 1 -1 1 1 1 1 -1 1 1 -1 1 -1 -1;
   -1 -1 1 -1 1 1 1 -1 -1 -1 -1 1 -1 -1 1 -1;1 -1 1 1 1 -1 1 1 1 -1 1 1 -1 1 -1 -1;
   1 -1 -1 -1 1 -1 -1 -1 -1 1 1 1 1 -1 -1 -1;-1 1 -1 -1 -1 1 1 1 1 -1 -1 -1 -1 1 -1 -1;
   -1 -1 -1 1 -1 -1 -1 1 -1 -1 -1 1 1 1 1 -1;1 -1 1 1 -1 1 -1 -1 -1 1 -1 -1 -1 1 -1 -1;
   -1 -1 -1 -1 -1 -1 1 1 1 -1 -1 1 1 -1 1 -1;1 1 -1 -1 1 1 1 1 -1 1 -1 1 -1 1 1 -1;
   1 -1 -1 -1 1 -1 1 1 -1 1 -1 -1 1 -1 -1 -1;-1 -1 -1 1 1 1 1 -1 1 1 1 -1 1 1 1 -1;
   1 -1 -1 -1 1 -1 -1 -1 1 -1 1 1 -1 1 -1 -1;1 -1 1 1 -1 1 -1 -1 1 -1 -1 -1 1 -1 -1 -1;
   1 1 -1 1 1 1 -1 1 -1 -1 -1 1 1 1 1 -1;-1 -1 -1 1 -1 -1 1 -1 -1 -1 1 -1 1 1 1 -1;
   -1 1 -1 -1 1 -1 1 1 1 -1 -1 -1 1 -1 -1 -1;-1 1 1 1 1 -1 -1 -1 1 -1 -1 -1 1 -1 -1 -1;
   -1 1 -1 1 -1 -1 -1 -1 -1 -1 1 1 -1 1 1 -1;1 1 -1 1 -1 -1 1 -1 -1 -1 1 -1 -1 -1 1 -1;
   1 -1 -1 -1 -1 1 1 1 1 -1 -1 -1 1 -1 -1 -1;-1 -1 1 1 -1 -1 -1 -1 -1 1 -1 1 -1 1 1 -1;
   -1 -1 1 -1 -1 -1 -1 1 1 1 1 -1 -1 -1 1 -1;-1 -1 1 -1 1 1 -1 1 1 1 1 -1 1 1 1 -1];
% Perform spreading and scrambling of data sequence
block=[];
for n=1:Nk;
   for q=1:Qk;
      s=c(k,1+mod((n-1)*Qk+q-1,Qk))*(i^(1+mod((n-1)*Qk+q-1,Qmax))*...
         v(cell+1,1+mod((n-1)*Qk+q-1,Qmax)));
      block(1,Qk*(n-1)+q)=do(1,n)*w(1,k)*s;
   end;
end;
% filter chipped sequence with root-raised cosine (RRC)
Delay=3;DataL=Nk*Qk;R=0.22;S=8;Fs=S*chiprate;Fd=chiprate;PropD=Delay/Fd;
x=block';
tx=[PropD:PropD+DataL-1]./Fd;
xreal=real(x);ximag=imag(x);
[yrealt,trealt]=rcosflt(xreal,Fd,Fs,'sqrt',R,Delay);% matching RRC filter
[yimagt,timagt]=rcosflt(ximag,Fd,Fs,'sqrt',R,Delay);% matching RRC filter
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Baseband Receiver
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
w1=conj(w);% to compensate for complex multiplier
probt=0;% ensure a clear probability of bit error counter
num=1;% number of iterations to run simulation
if M==8;floor=1.346;else;floor=.989;end;% set noise floor for AWGN
if Qk==1;pt=350;elseif Qk==2;pt=250;elseif Qk==4;pt=250;...
   elseif Qk==8;pt=250;else;pt=250;end;
for u=1:num;
   noise1=randn(length(yrealt),1)*sqrt(Qk/(2*log2(M)));% generate in-phase noise
   noise2=randn(length(yimagt),1)*sqrt(Qk/(2*log2(M)));% generate quadrature-phase
noise
   for r=1:pt;
      yrealc=yrealt+sqrt(1/10^floor)*noise1;% add noise to signal
      yimagc=yimagt-sqrt(1/10^floor)*noise2;% add noise to signal
      trealc=trealt;timagc=timagt;
      [yrealr,trealr]=rcosflt(yrealc,Fd,Fs,'sqrt/Fs');% matching RRC filter
      [yimagr,timagr]=rcosflt(yimagc,Fd,Fs,'sqrt/Fs');% matching RRC filter
      yrec=yrealr+sqrt(1/10^((r-100-1)/100))+i*yimagr;yrec=yrec';% add tone jamming
      L=length(yrealr)-length(yrealc)+1-S;% offset for time delay
      % sample RRC signal while despreading and descrambling
      block2=[];
      for n=1:Nk;
         for q=1:Qk;
            s=c(k,1+mod((n-1)*Qk+q-1,Qk))*((-i)^(1+mod((n-1)*Qk+q-1,Qmax))*...
               v(cell+1,1+mod((n-1)*Qk+q-1,Qmax)));
            block2(1,Qk*(n-1)+q)=yrec(1,S*(Qk*(n-1)+q)+L)*w1(1,k)*s;
         end;
      end;
      % use summer to recombine spread signal (could use integrator)
      block2=reshape(block2,Qk,Nk);
      if Qk==1;
         block2=block2;
      else;
         block2=sum(block2)/Qk;
      end;
      % compute arctangent to reconstruct MPSK signal
      if modu==1;
         for n=1:Nk;
            ytanr=atan2(imag(block2(1,n)),real(block2(1,n)));
            if ytanr < -3*pi/4;
               yr(1,n)=cos(9*pi/8)+i*sin(9*pi/8);
            elseif ytanr < -pi/2;
               yr(1,n)=cos(11*pi/8)+i*sin(11*pi/8);
            elseif ytanr < -pi/4;
               yr(1,n)=cos(13*pi/8)+i*sin(13*pi/8);
            elseif ytanr < 0;
```

```
                yr(1,n)=cos(15*pi/8)+i*sin(15*pi/8);
            elseif ytanr < pi/4;
                yr(1,n)=cos(pi/8)+i*sin(pi/8);
            elseif ytanr < pi/2;
                yr(1,n)=cos(3*pi/8)+i*sin(3*pi/8);
            elseif ytanr < 3*pi/4;
                yr(1,n)=cos(5*pi/8)+i*sin(5*pi/8);
            else;
                yr(1,n)=cos(7*pi/8)+i*sin(7*pi/8);
            end;
        end;
    else;
        for n=1:Nk;
            ytanr=atan2(imag(block2(1,n)),real(block2(1,n)));
            if ytanr < -3*pi/4;
                yr(1,n)=-1;
            elseif ytanr < -pi/4;
                yr(1,n)=-i;
            elseif ytanr < pi/4;
                yr(1,n)=1;
            elseif ytanr < 3*pi/4;
                yr(1,n)=i;
            else;
                yr(1,n)=-1;
            end;
        end;
    end;
    prob(1,r)=sum(abs(do-yr)>10^-10)/Nk;
  end;
  probt=probt+prob;
end;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% generate plots
r=1:1:pt;
if M==2;
  pe(1,r)=1/2*erfc(sqrt(10.^((r-100-1)/100)));
elseif M==4;
  pe(1,r)=erfc(sqrt(10.^((r-100-1)/100))).*(1-.25*erfc(sqrt(10.^((r-100-1)/100))));
else;
  pe(1,r)=erfc(sqrt(log2(M)*10.^((r-100-1)/100))*sin(pi/M));
end;
probj=probt/num;
pse2=1/2*1/2*erfc(sqrt(10^.989)*(1+sqrt(1/Qk*10.^-((r-100-1)/100))))+...
  1/2*1/2*erfc(sqrt(10^.989)*(1-sqrt(1/Qk*10.^-((r-100-1)/100))));
pse4=2*pse2.*(1-.5*pse2);
pse8=erfc(sqrt(log2(M)*10.^.989*sin(pi/M)));
```

```
pl(1,r)=1/2*erfc(sqrt(1./(10^-.989+2./Qk.*10.^-((r-100-1)/100))));
pl2=2*pl.*(1-.5*pl);
figure(1);
t=-10:.1:(pt-1)/10-10;
semilogy(t,pe,'r');hold on;
semilogy(t,probj);
if Qk==1;title(['P_s versus E_b/N_I, 8PSK and SF=1, E_b/N_0=9.89 dB']);
   semilogy(t,pse8,'g');
elseif Qk==2;title(['P_s versus E_b/N_I, QPSK and SF=2, E_b/N_0=9.89 dB']);
   semilogy(t,pse4,'g');semilogy(t,pl2,'m');
elseif Qk==4;title(['P_s versus E_b/N_I, QPSK and SF=4, E_b/N_0=9.89 dB']);
   semilogy(t,pse4,'g');semilogy(t,pl2,'m');
elseif Qk==8;title(['P_s versus E_b/N_I, QPSK and SF=8, E_b/N_0=9.89 dB']);
   semilogy(t,pse4,'g');semilogy(t,pl2,'m');
else;title(['P_s versus E_b/N_I, QPSK and SF=16, E_b/N_0=9.89 dB']);
   semilogy(t,pse4,'g');semilogy(t,pl2,'m');
end;
hold off;
if M==8;axis([-10 24.9 10^-5 10^0]);
else;axis([-10 14.9 10^-5 10^0]);
end;
xlabel('E_b/N_I');ylabel('BER');
```

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX D. MATLAB CODE FOR TD-SCDMA WITH BARRAGE JAMMING.

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% TD-SCDMA in additive white Gaussian noise (AWGN) with barrage jamming
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
clear
Qk=1;% Spreading Factor (SF)
if Qk==1;
   modu=1;M=8;% use 8PSK
else;
   modu=0;M=4;% use QPSK
end;
Qmax=16;% maximum spreading factor
k=1;% user number in a time slot
Kmax=16;% maximum number of users in a time slot
chiprate=1.28*10^6;% chiprate
Tc=1/chiprate;% chip duration = 1 / chiprate
cell=0;% cell number for scrambling code (could be random)
if Qk==1;
   c=[1]; % Orthogonal Variable Spreading Factor (OVSF)
   Nk=352; % Number of data points
   w=[1]; % weight multiplier
elseif Qk==2;
   c=[1 1;1 -1];Nk=176;w=[1 i];
elseif Qk==4;
   c=[1 1 1 1;1 1 -1 -1;1 -1 1 -1;1 -1 -1 1];Nk=88;w=[-i 1 i -1];
elseif Qk==8;
   c=[ 1 1 1 1 1 1 1 1;1 1 1 1 -1 -1 -1 -1;1 1 -1 -1 1 1 -1 -1;1 1 -1 -1 -1 -1 1 1;
      1 -1 1 -1 1 -1 1 -1;1 -1 1 -1 -1 1 -1 1;1 -1 -1 1 1 -1 -1 1;1 -1 -1 1 -1 1 1 -1];
   Nk=44;w=[1 i i -1 -i -1 -i 1];
else Qk==16;
   c=[ 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1;1 1 1 1 1 1 1 1 -1 -1 -1 -1 -1 -1 -1 -1;
      1 1 1 1 -1 -1 -1 -1 1 1 1 1 -1 -1 -1 -1;1 1 1 1 -1 -1 -1 -1 -1 -1 -1 -1 1 1 1 1;
      1 1 -1 -1 1 1 -1 -1 1 1 -1 -1 1 1 -1 -1;1 1 -1 -1 1 1 -1 -1 -1 -1 1 1 -1 -1 1 1;
      1 1 -1 -1 -1 -1 1 1 1 1 -1 -1 -1 -1 1 1;1 1 -1 -1 -1 -1 1 1 -1 -1 1 1 1 1 -1 -1;
      1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1;1 -1 1 -1 1 -1 1 -1 -1 1 -1 1 -1 1 -1 1;
      1 -1 1 -1 -1 1 -1 1 1 -1 1 -1 -1 1 -1 1;1 -1 1 -1 -1 1 -1 1 -1 1 -1 1 1 -1 1 -1;
      1 -1 -1 1 1 -1 -1 1 1 -1 -1 1 1 -1 -1 1;1 -1 -1 1 1 -1 -1 1 -1 1 1 -1 -1 1 1 -1;
      1 -1 -1 1 -1 1 1 -1 1 -1 -1 1 -1 1 1 -1;1 -1 -1 1 -1 1 1 -1 -1 1 1 -1 1 -1 -1 1];
   Nk=22;w=[-1 -i 1 1 i -1 -1 1 -i i 1 i -i -i i -1];
end
% Generate data sequence
temp=randint(1,Nk,M);
if modu==1;
```

```
  do=cos(pi/8+2*temp.*pi/8)+i*sin(pi/8+2*temp.*pi/8);% random 8PSK data sequence
else;
  do=i.^temp;% random QPSK data sequence
end;
% partial set of scrambling codes
v=[-1 1 -1 -1 -1 1 -1 -1 1 -1 1 1 -1 1 -1 -1;1 1 1 1 1 -1 1 -1 1 -1 -1 1 1 1 -1 -1;
  1 -1 1 1 1 -1 1 1 -1 1 1 1 1 -1 -1 -1;1 1 1 -1 -1 -1 -1 1 -1 -1 1 -1 -1 -1 1 -1;
  1 1 1 -1 -1 -1 -1 1 1 1 1 -1 1 1 1 -1;-1 1 1 -1 -1 -1 1 1 1 1 1 1 1 -1 1 -1;
  -1 1 -1 -1 -1 1 -1 -1 -1 1 1 1 1 -1 -1 -1;1 -1 1 -1 -1 -1 -1 -1 1 1 -1 -1 -1 1 1 -1;
  1 1 1 -1 -1 -1 1 -1 1 1 -1 1 1 1 1 -1;1 1 -1 1 1 1 1 -1 1 1 1 -1 -1 -1 1 -1;
  1 -1 1 -1 1 1 1 1 -1 -1 1 1 -1 1 1 -1;-1 1 1 1 1 -1 -1 -1 -1 1 -1 -1 -1 1 -1 -1;
  -1 -1 1 -1 -1 -1 1 -1 -1 -1 -1 1 1 1 1 -1;1 -1 1 1 1 -1 -1 -1 1 -1 -1 -1 -1 1 -1 -1;
  1 -1 -1 -1 -1 1 -1 -1 1 -1 1 1 1 -1 -1 -1;1 1 -1 -1 -1 1 1 -1 1 -1 1 -1 -1 -1 -1 -1;
  1 -1 -1 1 -1 1 -1 1 -1 -1 -1 -1 1 1 -1 -1;1 1 1 -1 1 1 1 -1 1 1 -1 1 -1 -1 1 -1;
  -1 1 1 1 -1 1 -1 -1 -1 1 -1 -1 1 -1 -1 -1;-1 1 -1 -1 1 -1 -1 -1 -1 1 1 1 -1 1 -1 -1;
  -1 -1 -1 -1 1 -1 1 -1 -1 1 1 -1 1 1 -1 -1;1 1 1 1 -1 -1 1 1 -1 1 1 -1 1 -1 1 -1;
  1 -1 -1 -1 -1 1 1 1 -1 1 -1 -1 -1 1 -1 -1;-1 1 1 1 -1 1 1 1 1 -1 1 1 -1 1 -1 -1;
  -1 -1 1 -1 1 1 1 -1 -1 -1 -1 1 -1 -1 1 -1;1 -1 1 1 1 -1 1 1 1 -1 1 1 -1 1 -1 -1;
  1 -1 -1 -1 1 -1 -1 -1 -1 1 1 1 1 -1 -1 -1;-1 1 -1 -1 -1 1 1 1 1 -1 -1 -1 -1 1 -1 -1;
  -1 -1 -1 1 -1 -1 -1 1 -1 -1 -1 1 1 1 1 -1;1 -1 1 1 -1 1 -1 -1 -1 1 -1 -1 -1 1 -1 -1;
  -1 -1 -1 -1 -1 -1 1 1 1 -1 -1 1 1 -1 1 -1;1 1 -1 -1 1 1 1 1 -1 1 -1 1 -1 1 1 -1;
  1 -1 -1 -1 1 -1 1 1 -1 1 -1 -1 1 -1 -1 -1;-1 -1 -1 1 1 1 1 -1 1 1 1 -1 1 1 1 -1;
  1 -1 -1 -1 1 -1 -1 -1 1 -1 1 1 -1 1 -1 -1;1 -1 1 1 -1 1 -1 -1 1 -1 -1 -1 1 -1 -1 -1;
  1 1 -1 1 1 1 -1 1 -1 -1 -1 1 1 1 1 -1;-1 -1 -1 1 -1 -1 1 -1 -1 -1 1 -1 1 1 1 -1;
  -1 1 -1 -1 1 -1 1 1 1 -1 -1 -1 1 -1 -1 -1;-1 1 1 1 1 -1 -1 -1 1 -1 -1 -1 1 -1 -1 -1;
  -1 1 -1 1 -1 -1 -1 -1 -1 -1 1 1 -1 1 1 -1;1 1 -1 1 -1 -1 1 -1 -1 -1 1 -1 -1 -1 1 -1;
  1 -1 -1 -1 -1 1 1 1 1 -1 -1 -1 1 -1 -1 -1;-1 -1 1 1 -1 -1 -1 -1 -1 1 -1 1 -1 1 1 -1;
  -1 -1 1 -1 -1 -1 -1 1 1 1 1 -1 -1 -1 1 -1;-1 -1 1 -1 1 1 -1 1 1 1 1 -1 1 1 1 -1];
% Perform spreading and scrambling of data sequence
block=[];
for n=1:Nk;
  for q=1:Qk;
    s=c(k,1+mod((n-1)*Qk+q-1,Qk))*(i^(1+mod((n-1)*Qk+q-1,Qmax))*...
      v(cell+1,1+mod((n-1)*Qk+q-1,Qmax)));
    block(1,Qk*(n-1)+q)=do(1,n)*w(1,k)*s;
  end;
end;
% filter chipped sequence with root-raised cosine (RRC)
Delay=3;DataL=Nk*Qk;R=0.22;S=8;Fs=S*chiprate;Fd=chiprate;PropD=Delay/Fd;
x=block';
tx=[PropD:PropD+DataL-1]./Fd;
xreal=real(x);ximag=imag(x);
[yrealt,trealt]=rcosflt(xreal,Fd,Fs,'sqrt',R,Delay);% matching RRC filter
[yimagt,timagt]=rcosflt(ximag,Fd,Fs,'sqrt',R,Delay);% matching RRC filter
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Baseband Receiver
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
w1=conj(w);% to compensate for complex multiplier
probt=0;% ensure a clear probability of bit error counter
num=1;% number of iterations to run simulation
if M==8;floor=1.346;else;floor=.989;end;% set noise floor for AWGN
pt=200;
for u=1:num;
   bnoise1=randn(length(yrealt),1)*sqrt(Qk/(2*log2(M)));% generate in-phase barrage noise
   bnoise2=randn(length(yimagt),1)*sqrt(Qk/(2*log2(M)));% generate quadrature-phase barrage noise
   noise1=randn(length(yrealt),1)*sqrt(Qk/(2*log2(M)));% generate in-phase noise
   noise2=randn(length(yimagt),1)*sqrt(Qk/(2*log2(M)));% generate quadrature-phase noise
   for r=1:pt;
      yrealc=yrealt+sqrt(1/(10^((r-1)/100)))*bnoise1+sqrt(1/10^floor)*noise1;
      yimagc=yimagt-sqrt(1/(10^((r-1)/100)))*bnoise2-sqrt(1/10^floor)*noise2;
      trealc=trealt;timagc=timagt;
      [yrealr,trealr]=rcosflt(yrealc,Fd,Fs,'sqrt/Fs');% matching RRC filter
      [yimagr,timagr]=rcosflt(yimagc,Fd,Fs,'sqrt/Fs');% matching RRC filter
      yrec=yrealr+i*yimagr;yrec=yrec';
      L=length(yrealr)-length(yrealc)+1-S;% offset for time delay
      % sample RRC signal while despreading and descrambling
      block2=[];
      for n=1:Nk;
         for q=1:Qk;
            s=c(k,1+mod((n-1)*Qk+q-1,Qk))*((-i)^(1+mod((n-1)*Qk+q-1,Qmax))*...
               v(cell+1,1+mod((n-1)*Qk+q-1,Qmax)));
            block2(1,Qk*(n-1)+q)=yrec(1,S*(Qk*(n-1)+q)+L)*w1(1,k)*s;
         end;
      end;
      % use summer to recombine spread signal (could use integrator)
      block2=reshape(block2,Qk,Nk);
      if Qk==1;
         block2=block2;
      else;
         block2=sum(block2)/Qk;
      end;
      % compute arctangent to reconstruct MPSK signal
      if modu==1;
         for n=1:Nk;
            ytanr=atan2(imag(block2(1,n)),real(block2(1,n)));
            if ytanr < -3*pi/4;
               yr(1,n)=cos(9*pi/8)+i*sin(9*pi/8);
            elseif ytanr < -pi/2;
               yr(1,n)=cos(11*pi/8)+i*sin(11*pi/8);
```

```
         elseif ytanr < -pi/4;
            yr(1,n)=cos(13*pi/8)+i*sin(13*pi/8);
         elseif ytanr < 0;
            yr(1,n)=cos(15*pi/8)+i*sin(15*pi/8);
         elseif ytanr < pi/4;
            yr(1,n)=cos(pi/8)+i*sin(pi/8);
         elseif ytanr < pi/2;
            yr(1,n)=cos(3*pi/8)+i*sin(3*pi/8);
         elseif ytanr < 3*pi/4;
            yr(1,n)=cos(5*pi/8)+i*sin(5*pi/8);
         else;
            yr(1,n)=cos(7*pi/8)+i*sin(7*pi/8);
         end;
      end;
   else;
      for n=1:Nk;
         ytanr=atan2(imag(block2(1,n)),real(block2(1,n)));
         if ytanr < -3*pi/4;
            yr(1,n)=-1;
         elseif ytanr < -pi/4;
            yr(1,n)=-i;
         elseif ytanr < pi/4;
            yr(1,n)=1;
         elseif ytanr < 3*pi/4;
            yr(1,n)=i;
         else;
            yr(1,n)=-1;
         end;
      end;
   end;
   prob(1,r)=sum(abs(do-yr)>10^-10)/Nk;
 end;
 probt=probt+prob;
end;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% generate plots
r=1:pt;
pe2=1/2*erfc(sqrt(10.^((r-1)/100)));
pe4=2*pe2.*(1-.5.*pe2);
pe8=erfc(sqrt(log2(M)*10.^((r-1)/100))*sin(pi/M));
peb(1,r)=erfc(sqrt(log2(M).*(10.^-((r-1)/100)+10^-.989).^-1)*sin(pi/M));
probj=probt/num;
figure(1);
t=0:.1:(pt-1)/10;
if M==4;semilogy(t,pe4,'r');else;semilogy(t,pe8,'r');end;
hold on;
```

```
semilogy(t,peb,'g');semilogy(t,probj,'b');hold off;
axis([0 20 10^-6 10^0]);
if Qk==1;title(['P_s versus E_b/N_I, 8PSK and SF=1, E_b/N_0=9.89 dB']);
elseif Qk==2;title(['P_s versus E_b/N_I, QPSK and SF=2, E_b/N_0=9.89 dB']);
elseif Qk==4;title(['P_s versus E_b/N_I, QPSK and SF=4, E_b/N_0=9.89 dB']);
elseif Qk==8;title(['P_s versus E_b/N_I, QPSK and SF=8, E_b/N_0=9.89 dB']);
else;title(['P_s versus E_b/N_I, QPSK and SF=16, E_b/N_0=9.89 dB']);
end;
xlabel('E_b/N_I');ylabel('BER');
```

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF REFERENCES

[1] Lee, W. C.Y., "Lee's Essentials of Wireless Communications," McGraw-Hill, New York, NY, 2001.

[2] Rappaport, T.S., "Wireless Communications: Principles and Practice," Prentice Hall, Upper Saddle River, NJ, 2002.

[3] Federal Communications Commission (FCC), "Third Generation ("3G") Wireless," [http://www.fcc.gov/3G], April 2002.

[4] International Telecommunications Union, "IMT-2000 Radio Interface Specifications Approved in ITU Meeting in Helsinki," [http://www.itu.int/newsarchive/press/releases/1999/99-22.html#1], November 1999.

[5] Siemens AG, "Taking the Right Path Towards 3G," [http://www.siemens-mobile.de/btob/external/downloads/path_towards_3G.pdf], April 2002.

[6] Siemens AG, "White Paper, TD-SCDMA: the Solution for TDD bands," [http://www.siemens-mobile.de/btob/external/downloads/TD-SCDMA/TD_SCDMA_White_Paper.pdf], April 2002.

[7] Peterson, R., Ziemer, R., Borth, D., "Introduction to Spread-Spectrum Communications," Prentice Hall, Upper Saddle River, NJ, 1995.

[8] 3GPP TS 25.223 V5.0.0, "Spreading and modulation (TDD)", March 2002.

[9] Sklar, B., "Digital Communications: Fundamentals and Applications," Prentice Hall, Upper Saddle River, NJ, 2001.

[10] 3GPP TS 25.937 V4.1.0, "Low chip rate TDD lub/lur protocol aspects", September, 2001.

[11] 3GPP TS 25.928 V4.0.1, "1.28Mcps functionality for UTRA TDD Physical Layer", March, 2001.

[12] Kammerlander, K., "TD-SCDMA an advance approach for 3G Mobile Radio Systems", Publication 09-2001, Draft 01, September, 2001.

[13] 3GPP TS 25.224 V4.3.0, "Physical Layer Procedures (TDD)", December, 2001.

[14] Proakis, J., Manolakis, D., "Digital Signal Processing: Principles, Algorithms, and Applications", Prentice Hall, Upper Saddle River, NJ, 1996.

[15] Holma, H., “WCMDA for UMTS”, WCNC 2002 presentation, March 21, 2001.

[16] Proakis, J., “Digital Communications”, McGraw-Hill, New York, NY, 2001.

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center
   Ft. Belvoir, Virginia

2. Dudley Knox Library
   Naval Postgraduate School
   Monterey, California

3. National Security Agency, Attn: Bob Eubanks (GRSO/ATD)
   Applied Technologies Division
   Fort George G. Meade, Maryland

4. Cryptologic Research Laboratory, Attn: Nathan Beltz
   Naval Postgraduate School
   Monterey, California

5. Dr. R. Clark Robertson, Department of Electrical and Computer Engineering, Code EC/Rc
   Naval Postgraduate School
   Monterey, California

6. Dr. Tri T. Ha, Department of Electrical and Computer Engineering, Code EC/Ha
   Naval Postgraduate School
   Monterey, California

7. Dr. Jeffrey Knorr, Chairman, Department of Electrical and Computer Engineering.
   Naval Postgraduate School
   Monterey, California

8. Nick Triska
   16 March Pond Ln
   Monroe, Connecticut

9. Donald H. Paulson, Sr.
   29042 480th Ave
   Canton, South Dakota